

TRABAJO DE FIN DE GRADO



Análisis, diseño e implementación de un sistema para ayudar en el control de inventario.

Autor:

Sergio Gómez González

Tutor:

Alejandro Calderón Mateos



Agradecimientos

A Alex, por ser un genial profesor, magnífico tutor y mejor persona. Por esas “tutorías de 20 minutillos para unas dudas” que se acababan convirtiendo en conversaciones de más de 2 horas sobre cualquier tema. Por ayudarme a resolver los problemas con los que me he encontrado, siempre sacándome una gran sonrisa. ¡Eres muy grande!

A la práctica de TC que suspendimos antes siquiera de poder empezarla, y que nos mostró lo “duro” que iba a ser la universidad. ¡Quién lo hubiera pensado!

A “El Mani”, por hacer más interesante el ir a la universidad para acabar en esas reuniones legendarias, por hacer de “*An innocent box*” una utopía y a Charly, por haber sido nuestro *refrescagaznates* oficial. Si esto no sale... ¡siempre nos quedará el Banesto de Brunete!

A Maikel, por esas clases ilegales sobre seguridad en la biblioteca. ¡Tienes madera de profesor *madafaka*!

A Mr.Laheronsus, por esos *G-Switch*, *Crazy Mammoth*, *Zombies*, *Minecraft* y demás *demases* que hacían menos dura la universidad. ¡Ojo con el *creeper*!

A Jess, por darnos esas ganas de fiesta siempre en el momento que más lo necesitábamos.

A Rober y Litos por esos “*supertron*” legendarios. *Burkina Faso* sorprendió a todos con sus victorias. ¡Nos queda la espinita de jugarlo donde ya sabéis!

A Diego, sobre todo por esas tardes interminables con DSO. Por ese portátil que entró en hibernación tras más de 2 horas compilando el *kernel*. ¡Recuerda mirar detrás del arbusto!

A Ernesto, por esas tutorías clandestinas que me echaron un cable en los huesos que más se me atragantaban. Por esos duelos legendarios que aumentaban de forma insultante mi lista de victorias. ¡Algún día me ganarás! (legalmente).

A Sandra y Ana, compañeras de fatigas desde E.S.O, con las que he compartido toda la trayectoria académica, y anécdotas y cervezas a partes iguales. Hemos pasado juntos todos los momentos buenos y los malos, en lo académico y en lo personal. Es bonito ver como el *Pack de Fuenla* sigue unido tras más de 12 años. ¡Sois muy grandes!

A mi hermano, sobre todo por el día que me llevó la calculadora a mi último examen de la carrera, porque a pesar de llegar una hora y media antes, me di cuenta de que me la había olvidado cuando quedaban veinte minutos para empezar, y no me daba tiempo a volver a casa a por ella. ¡Te debo un *llévamelacalculadora*!

A mis padres, por permitirme y animarme a estudiar. Por haber confiado en mí y haberme ayudado a tomar decisiones importantes. Sin ellos jamás lo hubiera conseguido. GRACIAS.

A mi chica, por recoger la toalla cuando la he tirado y empujarme de nuevo al ring para acabar el combate cuando más cerca he estado de rendirme. Gran parte de la culpa de que pronto pueda decir “soy ingeniero” es tuya. MUCHAS GRACIAS.

Y en general, a todos los que de una manera u otra han formado parte de mi vida durante este periodo al que pongo punto y final, ya que para bien o para mal, me han hecho crecer como persona y evolucionar en lo que soy ahora.



Índice de contenidos

Índice de figuras.....	7
Índice de tablas	8
Glosario de términos	12
Acrónimos.....	12
Definiciones	13
1 Introducción	18
1.1 Motivación.....	18
1.2 Objetivos	19
1.3 Organización del documento.....	20
2 Estado de la cuestión.....	21
2.1 Soluciones similares.....	21
2.1.1 GLPI	21
2.1.2 Scanpet.....	24
2.2 Solución propuesta	25
2.2.1 iNventariator	25
2.3 Tabla comparativa	25
2.4 Elementos utilizados.....	27
2.4.1 Phonegap	27
2.4.2 jQuery	33
2.4.3 jQuery Mobile	35
2.4.4 Web SQL	37
2.4.5 Plugin LazarSoft QR	39
2.4.6 Cifrado.....	41
3 Análisis, diseño, implementación e implantación	44
3.1 Introducción.....	44
3.1.1 Metodología utilizada.....	44
3.1.1.1 Metodologías Pesadas: Métrica V3.....	44
3.1.1.2 Metodologías Ágiles: Scrum.....	45
3.1.2 Ciclo de vida	45
3.1.2.1 Planificación del sprint.....	47
3.1.2.2 Seguimiento del sprint	47
3.1.2.3 Revisión del sprint.....	47
3.1.2.4 Retrospectiva del sprint	47
3.1.3 Roles.....	48
3.1.3.1 Product Owner o propietario del producto.....	48



3.1.3.2	Scrum Master o facilitador	48
3.1.3.3	Equipo de desarrollo	48
3.1.3.4	Stakeholders	48
3.1.4	Marco regulador	49
3.2	Análisis.....	49
3.2.1	Casos de uso.....	49
3.2.1.1	Diagramas de casos de uso.....	49
3.2.1.2	Definición textual de los casos de uso	51
3.2.2	Requisitos.....	62
3.2.2.1	Requisitos funcionales.....	62
3.2.2.2	Requisitos no funcionales.....	69
3.2.1	Matriz de trazabilidad Casos de Uso – Requisitos.....	71
3.3	Diseño.....	72
3.3.1	Diagramas de navegación	72
3.3.2	Esquema de la base de datos.....	73
3.3.2.1	Tabla USERS.....	73
3.3.2.2	Tabla COMPUTERS.....	74
3.3.2.3	Tabla LOG.....	74
3.3.3	Diseño de las pruebas	75
3.3.3.1	Matriz de trazabilidad Pruebas – Requisitos.....	81
3.4	Implementación	82
3.5	Implantación.....	84
3.5.1	Instalación en sistema Android	84
3.5.2	Manual de usuario	85
4	Planificación del proyecto	86
4.1	Planificación inicial.....	86
4.2	Desarrollo real	89
4.3	Comparativa entre la planificación inicial y el desarrollo real	92
5	Medios técnicos empleados para el proyecto.....	94
6	Presupuesto	96
6.1	Metodología de estimación de costes	96
6.2	Presupuesto inicial	97
6.2.1	Costes asociados a Recursos Humanos.....	97
6.2.2	Costes asociados a herramientas hardware.....	98
6.2.3	Costes asociados a herramientas software.....	98
6.2.4	Costes asociados a fungibles	99
6.2.5	Costes asociados a dietas y costes de viaje	99
6.2.6	Total costes directos.....	100



6.2.7	Total costes indirectos.....	101
6.2.8	Estimación inicial de costes	101
6.2.9	Presupuesto a entregar al cliente	102
6.3	Coste final del proyecto.....	103
6.3.1	Costes asociados a Recursos Humanos.....	103
6.3.2	Costes asociados a herramientas hardware.....	104
6.3.3	Costes asociados a herramientas software.....	104
6.3.4	Costes asociados a fungibles	104
6.3.5	Costes asociados a dietas y costes de viaje	104
6.3.6	Total costes directos.....	105
6.3.7	Total costes indirectos.....	106
6.3.8	Coste real del proyecto.....	106
6.4	Análisis de la desviación	107
7	Conclusiones y trabajos futuros.....	108
7.1	Conclusiones	108
7.1.1	Conclusiones relativas al sistema desarrollado	108
7.1.2	Conclusiones relativas a la planificación y el presupuesto.....	109
7.1.2.1	Problemas de planificación.....	109
7.1.2.2	Escenario alternativo en el desarrollo del proyecto	109
7.1.2.2.1	Planificación Alternativa.....	110
7.1.2.2.2	Costes alternativos del proyecto.....	111
7.1.3	Conclusiones personales	112
7.2	Trabajos futuros.....	114
7.2.1	Actualización a la última versión de Phonegap	114
7.2.2	Actualización a la última versión de jQuery y jQuery Mobile.....	114
7.2.3	Importación y exportación de listados a CSV	115
7.2.4	Exportación de listados a PDF	115
7.2.5	Sincronización entre diferentes dispositivos	115
7.2.6	Multilenguaje.....	115
7.2.7	Inventario de otros elementos informáticos.....	115
7.2.8	Mensajería interna.....	116
7.2.9	Opciones de accesibilidad.....	116
7.2.10	Diferentes roles para los usuarios.....	116
7.2.11	Estadísticas.....	116
7.2.12	Otros ámbitos de la aplicación	116
7.2.13	Mejora continua	116



8	Referencias	117
Anexo I: Manual de Usuario		119
1	Introducción	120
2	Pantalla de login	120
3	Pantalla principal.....	121
4	Máquinas	122
4.1	Listado de máquinas	122
4.1.1	Detalles de una máquina	123
4.1.2	Modificación de máquinas	124
4.2	Añadir una nueva máquina	125
4.3	Máquinas solicitadas.....	126
5	Usuarios.....	127
5.1	Listado de usuarios.....	127
5.1.1	Detalles de un usuario	128
5.1.2	Modificación de usuarios	129
5.2	Añadir un nuevo usuario	130
6	Log de la base de datos	131
7	Administración	132
8	Códigos QR	133
9	FAQs	135
Anexo II: Informe de pruebas.....		136
1	Informes de los casos de prueba	137
2	Conclusiones.....	151



Índice de figuras

Ilustración 1 - GLPI: Número excesivo de botones poco intuitivos.....	21
Ilustración 2 - GLPI: Cantidad de información excesiva	22
Ilustración 3 - GLPI: Cantidad de información excesiva II.....	23
Ilustración 4 - GLPI: Cabecera de la pantalla principal	23
Ilustración 5 - GLPI: Menú desplegable pantalla principal	23
Ilustración 6 - GLPI: Menú horizontal en la pantalla principal.	23
Ilustración 7 - GLPI: Menú al pulsar el botón 3 en la ilustración 4.	24
Ilustración 8 - Comparativa cuota de mercado	28
Ilustración 9 - Comparación de uso entre frameworks	33
Ilustración 10 - Ciclo de vida.....	46
Ilustración 11 - Diagrama de casos de uso genérico	49
Ilustración 12 - Diagrama de casos de uso para el control de máquinas.....	50
Ilustración 13 - Diagrama de casos de uso para el control de usuarios	50
Ilustración 14 - Diagrama de casos de uso para la administración	51
Ilustración 15 - Diagrama de navegación de ventanas.....	72
Ilustración 16 - Model ER de la base de datos	73
Ilustración 17 - Matriz de trazabilidad Pruebas de Aceptación y Requisitos.....	81
Ilustración 18 - Esquema de cifrado de contraseña.....	84
Ilustración 19 - Diagrama de Gantt de la planificación inicial del proyecto	88
Ilustración 20 - Diagrama de Gantt del desarrollo real del proyecto	91
Ilustración 21 - Pantalla de login.....	120
Ilustración 22 - Pantalla principal de la aplicación	121
Ilustración 23 - Pantalla Máquinas: Listado	122
Ilustración 24 - Pantalla Máquinas: Detalles de la máquina	123
Ilustración 25 - Pantalla Máquinas: Modificar máquina	124
Ilustración 26 - Pantalla Máquinas: Añadir una máquina	125
Ilustración 27 - Pantalla Máquinas: Máquinas solicitadas	126
Ilustración 28 - Pantalla Usuarios: Listado de usuarios	127
Ilustración 29 - Pantalla Usuarios: Detalles de usuario.....	128
Ilustración 30 - Pantalla Usuarios: Modificar usuario.....	129
Ilustración 31 - Pantalla Usuarios: Añadir usuario	130
Ilustración 32 - Log de la base de datos	131
Ilustración 33 - Panel de administración	132
Ilustración 34 - Captura de códigos QR.....	133
Ilustración 35 - Diagrama escaneo de códigos QR.....	134
Ilustración 36 - Pantalla de preguntas frecuentes.....	135



Índice de tablas

Tabla 1 - Acrónimos.....	12
Tabla 2 - Definiciones	15
Tabla 3 - Tabla comparativa soluciones.....	25
Tabla 4 - Tabla comparativa cuota de mercado.....	28
Tabla 5 - Tabla comparativa de sistemas	29
Tabla 6 - Tabla comparativa de lenguajes.....	31
Tabla 7 - Tabla comparativa de frameworks.....	34
Tabla 8 - Tabla comparativa de soporte estimado frameworks.....	35
Tabla 9 - Tabla comparativa de frameworks para la interfaz.....	36
Tabla 10 - Tabla comparativa de opciones de almacenamiento	38
Tabla 11 - Tabla comparativa de opciones de decodificación de códigos QR.....	39
Tabla 12 - Tabla comparativa de algoritmos de cifrado	42
Tabla 13 - CU-001: Consultar listado máquinas	52
Tabla 14 - CU-002: Consultar detalles de máquina.....	52
Tabla 15 - CU-003: Solicitar máquina.....	53
Tabla 16 - CU-004: Modificar datos máquina.....	53
Tabla 17 - CU-005: Eliminar máquina.....	54
Tabla 18 - CU-006: Añadir máquina.....	54
Tabla 19 - CU-007: Consultar máquinas solicitadas	55
Tabla 20 - CU-008: Capturar código QR	56
Tabla 21 - CU-009: Consultar listado usuarios	56
Tabla 22 - CU-010: Consultar detalles de usuario	56
Tabla 23 - CU-011: Modificar datos usuario	57
Tabla 24 - CU-012: Eliminar usuario	57
Tabla 25 - CU-013: Añadir usuario	58
Tabla 26 - CU-014: Crear base de datos con máquinas de ejemplo.....	58
Tabla 27 - CU-015: Crear base de datos con usuarios de ejemplo	59
Tabla 28 - CU-016: Consultar log de la base de datos.....	59
Tabla 29 - CU-017: Invertir orden eventos log de la base de datos.....	60
Tabla 30 - CU-018: Borrar base de datos de usuarios	60
Tabla 31 - CU-019: Borrar log de la base de datos	61
Tabla 32 - CU-020: Borrar base de datos de máquinas.....	61
Tabla 33 - RF-001: Login.....	63
Tabla 34 - RF-002: Listar máquinas.....	63
Tabla 35 - RF-003: Añadir máquina.....	63
Tabla 36 - RF-004: Gestión de solicitudes	63
Tabla 37 - RF-005: Información máquina.....	64
Tabla 38 - RF-006: Expandir campos información máquina	64
Tabla 39 - RF-007: Eliminar máquina.....	64
Tabla 40 - RF-008: Solicitar máquinas	64
Tabla 41 - RF-009: Editar campos máquina	65
Tabla 42 - RF-010: Búsqueda de usuarios	65
Tabla 43 - RF-011: Búsqueda de usuarios	65



Tabla 44 - RF-012: Uso de códigos QR capturados por la cámara.....	65
Tabla 45 - RF-013: Uso de códigos QR existentes en el álbum.....	66
Tabla 46 - RF-014: Listar usuarios	66
Tabla 47 - RF-015: Añadir usuario	66
Tabla 48 - RF-016: Información usuario	66
Tabla 49 - RF-017: Eliminar usuario	67
Tabla 50 - RF-018: Editar campos usuario	67
Tabla 51 - RF-019: Histórico de los cambios en la base de datos.....	67
Tabla 52 - RF-020: Búsqueda en el histórico de cambios en la base de datos	67
Tabla 53 - RF-021: Invertir orden del histórico de cambios en la base de datos	68
Tabla 54 - RF-022: Borrar el histórico de cambios en la base de datos	68
Tabla 55 - RF-023: Crear base de datos de máquinas	68
Tabla 56 - RF-024: Borrar base de datos de máquinas	68
Tabla 57 - RF-025: Crear base de datos de usuarios	69
Tabla 58 - RF-026: Borrar base de datos de usuarios	69
Tabla 59 - RF-027: Logout.....	69
Tabla 60 - RNF-001: La aplicación se desarrollará en Phonegap	70
Tabla 61 - RNF-002: La interfaz de la aplicación se gestionará mediante jQuery Mobile	70
Tabla 62 - RNF-003: El soporte de almacenamiento será WebSQL.....	70
Tabla 63 - RNF-004: El decodificador QR se gestionará de manera interna	70
Tabla 64 - Matriz de trazabilidad Casos de Uso - Requisitos	71
Tabla 65 - PA-001	75
Tabla 66 - PA-002	75
Tabla 67 - PA-003	75
Tabla 68 - PA-004	76
Tabla 69 - PA-005	76
Tabla 70 - PA-006	76
Tabla 71 - PA-007	76
Tabla 72 - PA-008	76
Tabla 73 - PA-009	77
Tabla 74 - PA-010	77
Tabla 75 - PA-011	77
Tabla 76 - PA-012	77
Tabla 77 - PA-013	77
Tabla 78 - PA-014	78
Tabla 79 - PA-015	78
Tabla 80 - PA-016	78
Tabla 81 - PA-017	78
Tabla 82 - PA-018	78
Tabla 83 - PA-019	79
Tabla 84 - PA-020	79
Tabla 85 - PA-021	79
Tabla 86 - PA-022	79
Tabla 87 - PA-023	79
Tabla 88 - PA-024	80
Tabla 89 - PA-025	80



Análisis, diseño e implementación de un sistema para ayudar en el control de inventario.
Índice de tablas.

Tabla 90 - PA-026.....	80
Tabla 91 - PA-027.....	80
Tabla 92 - PA-028.....	80
Tabla 93 - Planificación inicial del proyecto.....	87
Tabla 94 - Desarrollo real del proyecto.....	90
Tabla 95 - Comparativa entre la planificación inicial y el desarrollo real.....	93
Tabla 96 - Medios técnicos empleados para el proyecto (software).....	94
Tabla 97 - Medios técnicos empleados para el proyecto (hardware).....	95
Tabla 98 - Presupuesto inicial: Costes asociados a Recursos Humanos.....	97
Tabla 99 - Presupuesto inicial: Costes asociados a herramientas hardware.....	98
Tabla 100 - Presupuesto inicial: Costes asociados a herramientas software.....	99
Tabla 101 - Presupuesto inicial: Costes asociados a fungibles.....	99
Tabla 102 - Presupuesto inicial: Costes asociados a dietas y costes de viaje.....	100
Tabla 103 - Presupuesto inicial: Total costes directos.....	100
Tabla 104 - Presupuesto inicial: Total costes indirectos.....	101
Tabla 105 - Presupuesto inicial: Estimación inicial de costes.....	101
Tabla 106 - Presupuesto inicial: Presupuesto a entregar al cliente.....	102
Tabla 107 - Presupuesto inicial: Costes asociados a Recursos Humanos.....	103
Tabla 108 - Coste final del proyecto: Costes asociados a fungibles.....	104
Tabla 109 - Coste final del proyecto: Costes asociados a dietas y costes de viaje.....	105
Tabla 110 - Coste final del proyecto: Total costes directos.....	105
Tabla 111 - Coste final del proyecto: Total costes indirectos.....	106
Tabla 112 - Coste final del proyecto: Coste real del proyecto.....	106
Tabla 113 - Análisis de la desviación de costes.....	107
Tabla 114 - Planificación alternativa.....	110
Tabla 115 - Costes alternativos del proyecto.....	111
Tabla 116 - Informe de prueba PA-001.....	137
Tabla 117 - Informe de prueba PA-002.....	138
Tabla 118 - Informe de prueba PA-003.....	138
Tabla 119 - Informe de prueba PA-004.....	139
Tabla 120 - Informe de prueba PA-005.....	139
Tabla 121 - Informe de prueba PA-006.....	140
Tabla 122 - Informe de prueba PA-007.....	140
Tabla 123 - Informe de prueba PA-008.....	141
Tabla 124 - Informe de prueba PA-009.....	141
Tabla 125 - Informe de prueba PA-010.....	142
Tabla 126 - Informe de prueba PA-011.....	142
Tabla 127 - Informe de prueba PA-012.....	143
Tabla 128 - Informe de prueba PA-013.....	143
Tabla 129 - Informe de prueba PA-014.....	144
Tabla 130 - Informe de prueba PA-015.....	144
Tabla 131 - Informe de prueba PA-016.....	145
Tabla 132 - Informe de prueba PA-017.....	145
Tabla 133 - Informe de prueba PA-018.....	146
Tabla 134 - Informe de prueba PA-019.....	146
Tabla 135 - Informe de prueba PA-020.....	147



Tabla 136 - Informe de prueba PA-021.....	147
Tabla 137 - Informe de prueba PA-022.....	148
Tabla 138 - Informe de prueba PA-023.....	148
Tabla 139 - Informe de prueba PA-024.....	149
Tabla 140 - Informe de prueba PA-025.....	149
Tabla 141 - Informe de prueba PA-026.....	150
Tabla 142 - Informe de prueba PA-027.....	150
Tabla 143 - Informe de prueba PA-028.....	151



Glosario de términos

Acrónimos

Acrónimo	Definición
API	Siglas de <i>Application Programming Interface</i> . Se trata de un conjunto de funciones y procedimientos que ofrece cierta biblioteca para ser utilizados por otro <i>software</i> como una capa de abstracción, de modo que el programador pueda beneficiarse de ello, evitándose el trabajo de programar todo desde el principio.
CSS	Siglas de <i>Cascading Style Sheets</i> . Hacen referencia a un lenguaje de hojas de estilos utilizado para describir la presentación semántica (el aspecto y formato) de un documento escrito en lenguaje de marcas (por ejemplo, HTML).
DPDS	Siglas de Dirección de Proyectos de Desarrollo Software.
HTML	Siglas de <i>HyperText Markup Language</i> . Se refiere al lenguaje de marcado para la elaboración de páginas web.
IT	Siglas de <i>Information Technology</i> .
IVA	Siglas de Impuesto sobre el Valor Añadido.
JEE	Siglas de <i>Java Enterprise Edition</i> . Es una plataforma de programación para desarrollar y ejecutar <i>software</i> de aplicaciones en Java.
JSP	Siglas de <i>JavaServer Pages</i> . Se trata de una tecnología que ayuda a los desarrolladores de <i>software</i> a crear páginas web dinámicas basadas en HTML o XML. Usa lenguaje de programación Java.
MDV	Siglas de Metodología de Desarrollo Visual.
PVP	Siglas de Precio de Venta al Público.
SDK	Siglas de <i>Software Development Kit</i> .
SO	Siglas de Sistema Operativo.
SQL	Siglas de <i>Structured Query Language</i> . Lenguaje declarativo de acceso a bases de datos relacionales, que permite especificar diversos tipos de operaciones en ellas.
TADS	Siglas de Técnicas Ágiles de Desarrollo Software.
TIW	Siglas de Tecnologías Informáticas para la Web.
UC3M	Siglas de Universidad Carlos III de Madrid.

Tabla 1 - Acrónimos



Definiciones

Término	Definición
Android	Sistema Operativo basado en Linux, que está diseñado principalmente para dispositivos móviles con pantalla táctil.
Bada	Sistema Operativo para móviles desarrollado por Samsung.
Blackberry OS	Sistema Operativo para móviles desarrollado por RIM para los dispositivos Blackberry.
Callback	Se trata de un tipo de función que se utiliza como argumento de otra función, de modo que cuando se llama a esta última, se ejecuta la primera. Puede ser utilizado para escribir funciones asíncronas, para poder conocer el resultado de la misma sin necesidad de estar esperando a que finalice su ejecución.
Checksum	Es una función <i>hash</i> cuyo propósito es proteger la integridad de los datos, verificando que no hay discrepancias entre los datos fidedignos y los datos reales.
Código QR	Siglas de Código de Respuesta Rápida (<i>Quick Response Code</i> en inglés). Se trata de un código bidimensional del estilo de los códigos de barras en los que se puede almacenar información en una matriz de puntos. Se caracteriza por los tres cuadrados que están en las esquinas y que permiten detectar la posición del código al lector. A diferencia de otros códigos similares como los códigos BIDI, son de código abierto y sus derechos de patente (propiedad de Denso Wave) no son ejercidos. Estos códigos tienen un gran potencial, ya que pueden almacenar más de 7000 caracteres numéricos, más de 4000 caracteres alfanuméricos, y casi 3000 bytes de información. Su uso cada vez está más extendido y se utiliza con multitud de propósitos: localización, configuraciones, tarjetas de contacto, links, etc. Como curiosidad, comentar que un solo código QR es capaz de almacenar toda una partida de ajedrez.
Eclipse	Eclipse es un IDE (siglas de <i>Integrated Development Environment</i>) abierto y extensible. Un IDE se compone de un conjunto de herramientas útiles para un desarrollador de <i>software</i> . Básicamente, un IDE cuenta con un editor de código, un compilador/intérprete y un depurador. Eclipse sirve como IDE Java (aunque también da soporte a otros lenguajes como C/C++, Cobol, Fortran, etc.).
Framework	Es una estructura conceptual y tecnológica de soporte definido, normalmente con artefactos o módulos de <i>software</i> concretos, que pueden servir de base para la organización y desarrollo de <i>software</i> . Típicamente puede incluir soporte de programas, bibliotecas y un lenguaje interpretado entre otras herramientas, para así ayudar a desarrollar y unir los diferentes componentes de un proyecto.



Función hash	Se trata de un tipo de función que tiene como entrada un conjunto de elementos (por ejemplo, cadenas de caracteres) y los convierte en una salida de rango finito, normalmente cadenas de longitud fija.
iOS	Sistema Operativo desarrollado por Apple para sus dispositivos móviles. Está basado en una variante del <i>Match Kernel</i> de Mac OS X. Tiene 4 capas de abstracción: la capa del núcleo del sistema operativo, la capa de “Servicios Principales”, la capa de “Medios de comunicación” y la capa de “Cocoa Touch”.
Javascript	Lenguaje de programación interpretado. Se define como orientado a objetos, está basado en prototipos, es imperativo, débilmente tipado y dinámico. Principalmente se utiliza en su forma del lado del cliente, es decir, se ejecuta en el navegador del cliente, no en un servidor.
jQuery	jQuery es una biblioteca de Javascript que permite facilitar la manera de interactuar con los documentos HTML, ofreciendo una serie de funcionalidades de una manera simplificada, permitiendo lograr grandes resultados en menos tiempo y líneas de código.
jQuery Mobile	Versión de jQuery adaptada para dispositivos móviles. Es un <i>plugin</i> del jQuery original y por tanto, requiere que este último esté también instalado para funcionar.
Open-source	Término con el que se conoce comúnmente al <i>software</i> desarrollado y distribuido libremente.
Phonegap	<i>Framework</i> utilizado para el desarrollo de aplicaciones móviles. Permite desarrollar <i>apps</i> para móviles utilizando herramientas genéricas como HTML5, CSS3 y Javascript.
Plugin	Complemento o extensión que es una aplicación que se “enchufa” a otra para aportarle nueva funcionalidad, generalmente muy específica.
Port	Se trata del proceso de adaptar <i>software</i> , por ejemplo, de un lenguaje a otro.
Refactoring	Técnica que consiste en la modificación del código fuente sin cambiar su comportamiento. No ha de arreglar errores ni añadir funcionalidad, y se utiliza simplemente para mejorar la comprensión del código, cambiar su estructura, eliminar código muerto, etc.
Rendering	Es el proceso de generar una imagen a partir de un modelo.
Smartphone	En español “teléfono inteligente”. Es el nombre comercial que se le ha dado a aquellos teléfonos que pueden utilizarse como un ordenador de bolsillo, es decir, que permiten realizar tareas propias de un ordenador: consultar el correo electrónico, multitarea, acceso a Internet (Wi-Fi o 3G/4G), funciones multimedia (foto, vídeo, audio), GPS, programas de ofimática...Además permiten instalar cientos de aplicaciones de terceros con una amplia variedad de propósitos.



Symbian	Sistema Operativo producto de la alianza de varias empresas de telefonía móvil, entre las que estaban Nokia, Sony Mobile, Samsung, Siemens, Benq, LG, Motorola, etc.
Windows Phone	Sistema Operativo móvil desarrollado por Microsoft, y sucesor de la plataforma Windows Mobile. Está enfocado al consumo generalista, en lugar del mercado empresarial, como era su predecesor. Su objetivo es competir directamente con Android e iOS.

Tabla 2 - Definiciones



Análisis, diseño e implementación de un sistema para ayudar en el control de inventario.
Resumen.

Resumen

En los tiempos que corren prácticamente todo está controlado mediante elementos informáticos. Además son imprescindibles para el desarrollo de las tareas diarias en las empresas y/o entidades.

El proyecto describe el análisis, diseño e implementación de un sistema para ayudar en el control de inventario, mejorando la eficiencia con la que éste es controlado, permitiendo de este modo ahorrar tiempo y dinero a las empresas y/o entidades en las que se utiliza el mismo.

Palabras clave: inventario, control, eficiencia, elementos informáticos.



Análisis, diseño e implementación de un sistema para ayudar en el control de inventario.
Abstract.

Abstract

Nowadays almost everything is controlled by computer devices. Furthermore, these devices are essential for the development of daily tasks in companies and/or entities.

This project describes the analysis, design and implementation of a helping system in inventory control, improving the efficiency with which it is controlled, allowing in this way to save time and money to the companies and/or entities that use it.

Key words: inventory, control, efficiency, computer devices.



1 Introducción

En este capítulo se explicarán las motivaciones que me han llevado a realizar este Trabajo de Fin de Grado. Posteriormente se enumerarán una serie de objetivos que se esperan alcanzar con la realización del mismo. Por último, se comentará brevemente la organización que sigue el documento.

1.1 Motivación

Hoy en día, prácticamente la totalidad de empresas y entidades (como por ejemplo administraciones, bibliotecas, universidades, hospitales, etc.) cuentan con elementos informáticos: ordenadores, ratones, teclados, monitores, cámaras web, lápices de memoria USB, discos duros externos, adaptadores, etc.

Estos elementos pueden utilizarse con multitud de propósitos: control y manejo de maquinaria, gestión de bases de datos, almacenamiento, enseñanza, programación, labores de ingeniería, edición de archivos multimedia, soportes ofimáticos, etc. Debido a múltiples causas (por ejemplo, diferentes turnos en una oficina, aulas informáticas, etc.), estos elementos han de ser compartidos, lo que conlleva que haya muchos usuarios potenciales involucrados en la utilización de los mismos.

En cualquiera de los casos mencionados anteriormente, el control de estos elementos puede hacerse algo tedioso en sí mismo.

Durante la realización de mis prácticas académicas en el departamento IT de una empresa de ingeniería, me di cuenta del caos que a veces supone el control de todo el parque informático: elementos que salen a oficinas del cliente (o que vuelven de ellas), elementos que se estropean, elementos nuevos, cambio de elementos a los usuarios, etc.

Las empresas y/o entidades pueden tener diferentes sistemas para controlar el inventario informático; algunos básicos como por ejemplo listados Excel, u otros algo más sofisticados, como por ejemplo bases de datos y herramientas dedicadas. El objetivo de cualquiera de estas herramientas es mantener sincronizado en todo momento el inventario con la situación real de los elementos inventariados.

En el caso de mi empresa, contaban con una herramienta *open-source* que permite el control del inventario. El origen del problema era que si no podías (por ejemplo, si te encontrabas sin conexión de red, o si te encontrabas trabajando fuera del ordenador durante un determinado tiempo) actualizar el mismo en el momento que ocurrían los cambios que sufren los ordenadores diariamente (cambios de nombre de la máquina, de usuario, de estado, de localización, etc.), corrías el riesgo de olvidarlo y que el inventario quedara desactualizado. Además, la interfaz no resultaba demasiado amigable por la gran cantidad de datos y opciones que mostraba, lo que podía ocasionar errores y pérdidas de tiempo innecesarias.



Por todo ello, pensé en realizar algo que facilitara estas tareas. Pensé que sería buena idea desarrollar una solución que permitiera el control del inventario de una forma más fácil a la que conocía, y que además, permitiera registrar los cambios en cualquier momento y en cualquier sitio, es decir, que mejore la movilidad.

Estos elementos, sobre todo los ordenadores, cuestan miles de euros a la empresa, por lo que un correcto control de los mismos es primordial, pero este control no debe afectar al desempeño de otras labores que afectan directamente a la producción de la empresa.

1.2 Objetivos

Por todo lo visto en el anterior apartado, los principales objetivos que se plantean con la realización del Trabajo de Fin de Grado son:

- Permitir a los usuarios actuar in situ, sin necesidad de estar delante de un ordenador. Es decir, que sea una aplicación “portátil”, para que se pueda actualizar el inventario desde cualquier lugar y en cualquier momento, justo cuando se tiene constancia del cambio, sin tener que esperar a llegar a un ordenador para realizarlos.
- Diseño intuitivo, para que los usuarios puedan usarla sin un proceso de aprendizaje previo. De este modo se podrán realizar las gestiones pertinentes sobre el inventario de una forma sencilla y eficaz.

De los que se deducen también estos objetivos:

- Agilizar el control del inventario informático en una empresa y/o entidad. Este control puede suponer un problema si no se hace de manera eficiente. Con la solución a proponer se pretende que dicho control sea un proceso más trivial, para poder reducir el tiempo invertido en el control del inventario y así poder dedicar un mayor tiempo a labores de más importancia para las actividades de la empresa.
- Mantener en todo momento una representación fiel de la realidad, y mostrar los datos de tal forma que facilite la toma de decisiones (por ejemplo, con el filtrado según diferentes criterios en las consultas, etc.). De este modo es más fácil conocer cuándo se tiene el stock necesario para cumplir con la demanda, o por el contrario, cuándo es necesario adquirir dispositivos para cubrir las necesidades.

El inventario informático que inicialmente se va a controlar con la aplicación desarrollada son los ordenadores. Esto es debido a que se trata del elemento cuya mala gestión provocaría un mayor coste a la empresa.



1.3 Organización del documento

En este apartado se resume la organización que sigue el presente documento:

En el capítulo Estado de la cuestión se explicarán las soluciones ya existentes al problema que se plantea en el capítulo de Motivación. Se hará una comparativa entre dichas soluciones y la solución propuesta en este Trabajo de Fin de Grado, justificando su elección. También se hablará de los elementos que han sido utilizados para la realización del mismo.

En el capítulo Análisis, diseño, implementación e implantación se mostrará todo lo relacionado con el desarrollo del *software* propiamente dicho. En primer lugar, en la Introducción se explicará la metodología que se ha seguido para el desarrollo del Trabajo de Fin de Grado, así como el marco regulador que afecta al mismo. En el apartado de Análisis se incluirá un catálogo de casos de uso y un catálogo de requisitos. En el apartado de Diseño se incluirá el diagrama de navegación de la aplicación, el esquema de la base de datos y el diseño de las pruebas. En el apartado de Implementación se explicarán aquellos detalles que ha costado más implementar. Por último, en el apartado de Implantación se incluirá una breve explicación para la implantación de la aplicación.

En el capítulo Planificación del proyecto se mostrará la planificación que inicialmente se ha hecho para la realización del Trabajo de Fin de Grado, incluyendo todas las fases del mismo, junto con una estimación del esfuerzo que se va a requerir en cada una de ellas. Posteriormente se mostrará el desarrollo real que finalmente ha tenido el mismo, y un breve estudio de las desviaciones producidas así como de las causas que la han provocado.

En el capítulo Medios técnicos empleados para el proyecto se mostrarán el *software* y *hardware* que ha sido utilizado durante la realización del proyecto.

En el capítulo Presupuesto se explicará la metodología que se ha utilizado para el cálculo de los costes asociados al proyecto. Se realizará una estimación inicial del presupuesto en función de la planificación hecha. Posteriormente se realizará un estudio de los gastos que finalmente ha conllevado la realización del proyecto. Por último se hará un breve estudio de la desviación producida así como de las causas que la han provocado.

En el capítulo Conclusiones y trabajos futuros se repasará que se hayan cumplido todos los objetivos marcados al inicio del Trabajo de Fin de Grado, se comentarán todos los problemas encontrados durante su realización, los conocimientos adquiridos durante las diferentes asignaturas de la carrera que han sido aplicados en el Trabajo de Fin de Grado, así como los conocimientos específicos adquiridos por cuenta propia durante la realización del mismo. Por último, se enumerarán una serie de trabajos futuros con el fin de mejorar la aplicación realizada.

En el capítulo Referencias se listarán aquellas referencias citadas durante el documento.



2 Estado de la cuestión

En este capítulo se hablará sobre soluciones ya existentes para el problema que trata de remediar la solución expuesta en este Trabajo de Fin de Grado. Se comentarán las características de cada una de las soluciones, para posteriormente realizar una comparación entre todas ellas. Finalmente se justificará la solución elegida.

2.1 Soluciones similares

2.1.1 GLPI

GLPI [1] era la solución utilizada en mi empresa para controlar todo el inventario informático: monitores, ratones, teclados y, principalmente, ordenadores.

Con dicha solución se tenía un control exhaustivo de todo el inventario. Por ejemplo, de un ordenador se podía conocer el nombre, número de serie, usuario, responsable, departamento, situación, estado, tipo, SO y *software* instalado, *hardware* disponible, red a la que estaba conectado, dirección IP, comentarios, etc.

A favor tenía el hecho de poder llevar un control exhaustivo de todo el inventario, permitiendo conocer casi cualquier detalle de cada uno de los elementos. Además, se trata de una solución *open-source*.

Por el contrario, puede resultar confuso el hecho de mostrar tanta información, botones poco intuitivos, etc., lo que conlleva una serie de aspectos negativos como pérdidas de tiempo y errores innecesarios.

A continuación se pueden observar algunos de los problemas citados anteriormente:

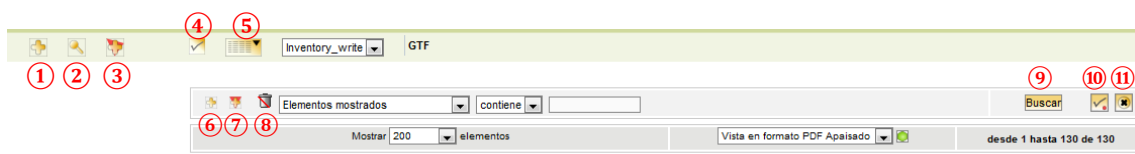


Ilustración 1 - GLPI: Número excesivo de botones poco intuitivos.

En el caso de la anterior ilustración, se puede observar un ejemplo de cómo el número de botones es excesivo y estos son poco intuitivos, pudiendo generar confusión al usuario. En la pantalla principal de gestión de ordenadores, el usuario se encuentra con 11 botones, 8 de los cuales están prácticamente pegados, y algunos idénticos entre ellos (por ejemplo, el ① y el ⑥, o el ③ y el ⑦), y poco intuitivos (no se intuye diferencia alguna entre el ① y el ③, por ejemplo). La funcionalidad que tienen los botones es la siguiente:



Análisis, diseño e implementación de un sistema para ayudar en el control de inventario. Capítulo II: Estado de la cuestión

- ①- Añadir nueva máquina.
- ②- Buscar.
- ③- Gestión de plantillas.
- ④- Cargar marcador.
- ⑤- Menú desplegable con las mismas opciones que en el menú.
- ⑥- Añadir criterio de búsqueda.
- ⑦- Añadir criterio global de búsqueda.
- ⑧- Ver los elementos borrados.
- ⑨- Buscar.
- ⑩- Registrar marcador.
- ⑪- Borrar.

Como se puede observar, la mayoría no son intuitivos, y algunos son redundantes.

En la siguiente ilustración, se puede observar un ejemplo de sobreinformación, en este caso, cuando queremos conocer el *hardware* relativo a una determinada máquina:

Componentes			
1	▼	Procesador	Intel(R) Pentium(R) 4 CPU 3.40GHz [1 core(s) x86]
1	▼	Procesador	Intel(R) Pentium(R) 4 CPU 3.40GHz [1 core(s) x86_64]
1	▼	Memoria RAM	Flash - SYSTEM ROM (No ECC)
1	▼	Memoria RAM	DDR2 - XMM1 (Multi-bit ECC)
1	▼	Memoria RAM	DDR2 - XMM3 (Multi-bit ECC)
1	▼	Memoria RAM	DDR2 - XMM2 (Multi-bit ECC)
1	▼	Disco duro	SAMSUNG HD160JJ
6	▼	Disco duro	WD Elements 1042 USB Device
2	▼	Disco duro	TOSHIBA MK127G5X USB Device
7	▼	Disco duro	Freecom USB Device
1	▼	Disco duro	SAMSUNG HD160JJ ATA Device
1	▼	Tarjeta de red	Broadcom NetXtreme Gigabit Ethernet - Teefer2 Miniport
1	▼	Tarjeta de red	Gigabit Ethernet Broadcom NetXtreme
1	▼	Lectoras	Lecteur de disquettes
1	▼	Lectoras	LITE-ON COMBO S0HC-4836K
1	▼	Lectoras	USB Device
1	▼	Lectoras	General USB Flash Disk USB Device
1	▼	Lectoras	Unidad de disquete
1	▼	Lectoras	MJYXIL 060ZVTE SCSI CdRom Device
1	▼	Lectoras	LITE-ON COMBO S0HC-4836K ATA Device
1	▼	Tarjeta gráfica	NVIDIA Quadro FX 1400
2	▼	Tarjeta gráfica	NumaraMirror.3.1.2-109421m
1	▼	Tarjeta gráfica	NVIDIA Quadro FX 1400 (Microsoft Corporation - WDDM)
1	▼	Tarjeta de sonido	Realtek High Definition Audio
1	▼	Tarjeta de sonido	Dispositivo de High Definition Audio
1	▼	Otros componentes	PASSWORD CLEAR (E49)
1	▼	Otros componentes	Mouse Port MOUSE
1	▼	Otros componentes	Audio Port CD-IN
1	▼	Otros componentes	Audio Port AUX-IN
1	▼	Otros componentes	None IDE
1	▼	Otros componentes	None FDD
1	▼	Otros componentes	SYS FAN
1	▼	Otros componentes	SPEAKER (SPK)
1	▼	Otros componentes	HOOD SENSE (HD SENSE)
1	▼	Otros componentes	HOOD LOCK
1	▼	Otros componentes	SECURITY
1	▼	Otros componentes	PCI FAN
1	▼	Otros componentes	Parallel Port ECP/EPP PARALLEL
1	▼	Otros componentes	Video Port VIDEO
1	▼	Otros componentes	SATA1
1	▼	Otros componentes	SATA2
1	▼	Otros componentes	Serial Port 16550A Compatible COM A
1	▼	Otros componentes	Serial Port 16550A Compatible SERIAL B
1	▼	Otros componentes	USB USB 1
1	▼	Otros componentes	USB USB 2
1	▼	Otros componentes	USB USB 3
1	▼	Otros componentes	USB USB 4
1	▼	Otros componentes	USB USB 5
1	▼	Otros componentes	USB USB 6
1	▼	Otros componentes	Keyboard Port KEYBOARD
1	▼	Otros componentes	Audio Port REAR HEADPHONE/LINEOUT
1	▼	Otros componentes	Network Port RJ45
1	▼	Otros componentes	CPU FAN
1	▼	Otros componentes	SATA0
1	▼	Otros componentes	USB USB 7
1	▼	Otros componentes	USB USB 8
1	▼	Otros componentes	Audio Port REAR LINE IN
1	▼	Otros componentes	Audio Port FRONT HEADPHONE/LINEOUT & MIC
1	▼	Otros componentes	Multibay
1	▼	Otros componentes	Audio Port REAR MIC
1	▼	Otros componentes	SATA3

Actualizar

Añadir un nuevo componente:

Añadir

Ilustración 2 - GLPI: Cantidad de información excesiva

Como se puede observar, la cantidad de información que es mostrada al usuario es excesiva, sin tener la mayoría además la menor relevancia.



Análisis, diseño e implementación de un sistema para ayudar en el control de inventario. Capítulo II: Estado de la cuestión

En la siguiente ilustración¹ se puede observar otro ejemplo de sobreinformación, en este caso en el listado de máquinas del inventario.

Ilustración 3 - GLPI: Cantidad de información excesiva II

Como se puede observar, la forma en la que se muestra la información no es la más amigable para el usuario, y como se ha comentado en el apartado de Motivación, el hecho de no poder actualizar siempre el inventario en el momento, produce que algunos datos no estén completos (marcados en rojo en la Ilustración 3).

Por último, como se puede ver en las siguientes ilustraciones, los menús en la pantalla principal son redundantes:



Ilustración 4 - GLPI: Cabecera de la pantalla principal

Ordenadores	Proveedores	Notas
Monitores	Contactos	Informes
Software	Documentos	
Electrónica de red		
Periféricos		
Impresoras		
Estados		

Ilustración 5 - GLPI: Menú desplegable pantalla principal

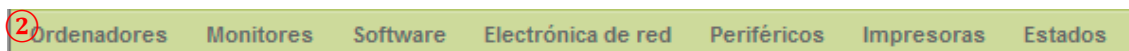


Ilustración 6 - GLPI: Menú horizontal en la pantalla principal.

¹ Los datos mostrados en dicha ilustración han sido pixelados para preservar la privacidad de los mismos.

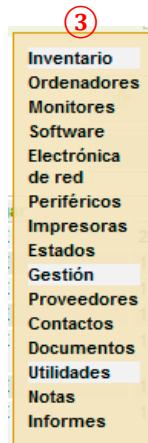


Ilustración 7 - GLPI: Menú al pulsar el botón 3 en la ilustración 4.

2.1.2 Scanpet

Tras el análisis que se hizo de GLPI, se detectaron dos importantes carencias: portabilidad y usabilidad, por lo que la solución propuesta se centrará en las mismas. A la hora de trabajar en el problema de la portabilidad, se ha decidido que la aplicación a desarrollar sea una aplicación para dispositivos móviles. Una vez se tomó dicha decisión, se comprobó si había alguna aplicación móvil existente que solucionara los problemas de usabilidad encontrados. Se encontró una aplicación llamada Scanpet [2]. Dicha aplicación fue descargada para poder realizar un estudio de la misma. Nada más ejecutarla por primera vez, puede observarse el primer problema encontrado. Scanpet nos da la opción de utilizar la aplicación de tres modos diferentes:

- Modo prueba: gratuito, pero sin acceso a todas las funciones.
- Modo licencia diaria: gratuito, con acceso a todas las funciones, pero con el requerimiento de ver un anuncio.
- Modo total: comprar la aplicación por un precio de 1,60€.

Mejorado el problema de la portabilidad y teniendo como objetivo mejorar la usabilidad, el tener publicidad invasiva en la pantalla, o el tener que ver un anuncio para poder utilizar la totalidad de la aplicación no parece lo más lógico. Además, se han detectado otros problemas en la usabilidad de la interfaz, como por ejemplo la muestra de formularios poco ordenados al usuario.

Otro problema que se ha detectado es que los datos son almacenados en un documento Excel, en lugar de una base de datos. Los documentos Excel están bien en el caso de querer exportar los datos a un documento “legible” desde fuera de la aplicación, o por ejemplo, para poder importar una lista a la aplicación, pero no como sistema de soporte de datos de la aplicación.



2.2 Solución propuesta

2.2.1 iNventariator

Con GLPI se tiene un control exhaustivo del inventario, pero como se ha comentado anteriormente, tiene algunas carencias importantes que el sistema que propongo busca solucionar. Las dos principales carencias detectadas son: portabilidad y usabilidad, por lo que iNventariator se centrará en las mismas. A la hora de trabajar en el problema de la portabilidad, se ha decidido que iNventariator sea una aplicación para dispositivos móviles, como ya se ha comentado en el anterior apartado. Para mejorar el problema de la usabilidad que también presentaba Scanpet, se han utilizado estándares [3] para dichos dispositivos móviles, de modo que la aplicación sea intuitiva y de un manejo sencillo. Se podría resumir la esencia de la aplicación con la siguiente frase: *“Tener la información que se necesita siempre a mano”*.

2.3 Tabla comparativa

A continuación se puede observar una tabla comparando las características de las diferentes soluciones vistas anteriormente:

Característica Solución	1	2	3	4	5
GLPI	M	A	B	<input checked="" type="checkbox"/>	A+
Scanpet	M+	M	A	<input checked="" type="checkbox"/>	A-
iNventariator	A	B+	A	<input checked="" type="checkbox"/>	M+

Tabla 3 - Tabla comparativa soluciones

1- Usabilidad.

- A: Alta.
- M: Media.
- B: Baja.

2- Periodo de aprendizaje.

- A: Alto.
- M: Medio.
- B: Bajo.

3- Movilidad.



4- Multiusuario.

5- Control del inventario.

- A: Alto.
- M: Medio.
- B: Bajo.

Seguidamente se explican los valores asignados en la anterior tabla a cada una de las características para cada una de las soluciones.

Usabilidad:

- GLPI: por motivos que ya se han explicado anteriormente, la usabilidad en GLPI no es demasiado buena: muestra demasiada información, botones poco intuitivos y una gran desorganización. A pesar de esto, tras un buen entrenamiento, se puede utilizar de forma más o menos efectiva, por lo que el valor que le ha sido asignado ha sido media.
- Scanpet: si bien es cierto que la usabilidad mejora respecto a GLPI, sigue habiendo ciertos detalles que hacen que la usabilidad no sea la mejor, como se ha comentado en la sección 2.1.2. Por ello el valor que se le ha asignado ha sido media-alta.
- iNventariator: en este caso la interfaz se ha hecho atendiendo a criterios de usabilidad, de forma que todo sea lo más intuitivo posible, y de modo que el usuario pueda utilizar la aplicación en su totalidad sin necesidad de un entrenamiento previo. Por ello, el valor que le ha sido asignado ha sido alta.

Periodo de aprendizaje:

- GLPI: siguiendo con el razonamiento expuesto en la anterior característica, esta falta de usabilidad hace que el periodo de aprendizaje sea alto. La mayor parte de la funcionalidad ha de ser aprendida, ya que no es muy intuitivo. Por ello el valor que se le ha asignado ha sido alto.
- Scanpet: el funcionamiento en sí (añadir y consultar elementos al inventario) es sencillo, y su periodo de aprendizaje es bajo. Sin embargo, la aplicación permite al usuario modificar muchos parámetros de configuración, algunos poco intuitivos, lo que puede infundir errores en el tratamiento de los datos. Por ello, el valor que se le ha asignado ha sido medio.
- iNventariator: en este caso la interfaz ha sido diseñada de tal modo que sea lo más intuitivo posible, permitiendo al usuario utilizar la aplicación sin necesidad de un periodo de aprendizaje. Por ello el valor que se le ha asignado ha sido bajo-medio. El motivo por el que no se le ha asignado el valor bajo es porque puede haber personas que no estén familiarizados con un entorno táctil, si bien es cierto que hoy en día prácticamente la totalidad de los teléfonos móviles son de este tipo.



Movilidad:

- GLPI: en el caso de GLPI se trata de una aplicación web para escritorio, que sólo puede ser utilizada desde un ordenador, por lo que la movilidad es bastante limitada. Por ello el valor que se le ha asignado ha sido baja.
- Scanpet: en este caso, al tratarse de una aplicación para dispositivos móviles, se puede utilizar desde cualquier lugar y en cualquier momento, por lo que está dotada de una gran movilidad. Por ello el valor que se le ha asignado es alta.
- iNventariator: en este caso ocurre lo mismo que en el anterior, y por lo tanto, se le ha asignado también el valor alta.

Multiusuario:

En el caso de GLPI e iNventariator la aplicación soporta diferentes usuarios. En el caso de Scanpet se puede cambiar el *id_usuario* para que cambie el valor que se inserta en los Excel, pero la aplicación no tiene la opción multiusuario como tal.

Control del inventario:

- GLPI: en este caso, GLPI permite llevar un control exhaustivo del inventario, no ciñéndose únicamente a ordenadores, si no que permite hacer inventario de casi cualquier cosa: monitores, teclados, ratones...Además, sobre los ordenadores ofrece más información, por ejemplo: *software* instalado, periféricos, características, etc. Por ello el valor que se le ha sido asignado ha sido muy alto.
- Scanpet: en este caso se puede hacer inventario de cualquier cosa, dejando en manos del usuario la organización que seguirá el fichero donde se almacenan los datos. El valor que se le ha asignado ha sido alto-medio.
- iNventariator: en el caso de iNventariator el control del inventario está inicialmente centrado únicamente en los ordenadores, permitiendo un buen control del inventario, pero sin ser tan detallado como GLPI ni tan personalizable como Scanpet. Por ello el valor que se le ha asignado ha sido medio-alto.

2.4 Elementos utilizados

2.4.1 Phonegap

Antes de comenzar con la realización de la aplicación para dispositivos móviles del Trabajo de Fin de Grado, se barajaron diferentes sistemas para los que desarrollarla. Se investigó sobre la cuota de mercado de los mismos [4], como queda resumido en la siguiente tabla:



SO Smartphone	Q2'12	Q3 '12	Q2'13	Q3 '13
Android	69,50%	75%	79,50%	81,3%
iOS	16,60%	15,6%	13,60%	13,4%
Windows Phone	3,60%	2,1%	3,90%	4,1%
Otros	10,40%	3%	3%	0,2%

Tabla 4 - Tabla comparativa cuota de mercado

A continuación se puede observar dicha comparativa a lo largo del tiempo de un modo más visual:

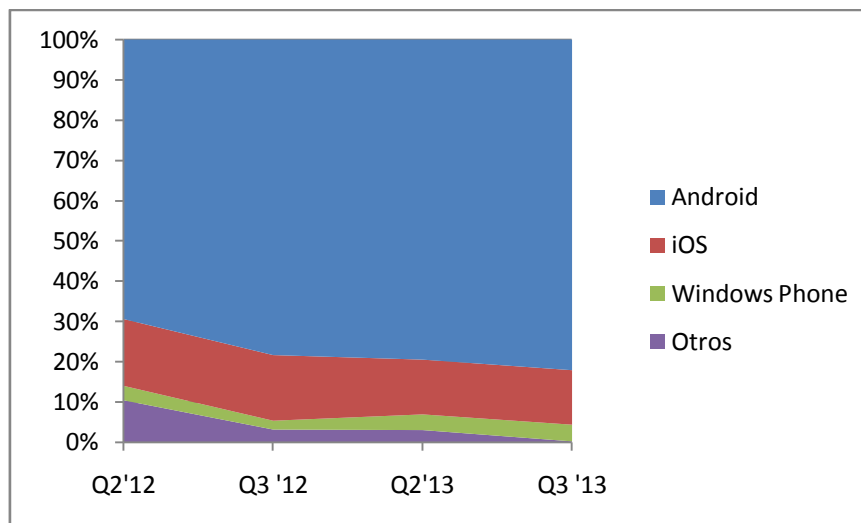


Ilustración 8 - Comparativa cuota de mercado

El objetivo del estudio visto previamente es el de seleccionar la plataforma primaria para la que se va a desarrollar la aplicación. Inicialmente se busca que funcione en dicha plataforma, para luego añadir otras adicionales. Este estudio busca dar respuesta a la pregunta: ¿Cuál debería ser la plataforma primaria para la que desarrollar la aplicación?

Tras finalizar el mismo, se descartó realizar inicialmente la aplicación para Windows Phone u otros sistemas como por ejemplo Symbian o Blackberry OS debido a sus bajas cuotas de mercado. En el caso de iOS fue descartado por varios motivos: menor cuota de mercado (y observando la tendencia que habían seguido durante 1 año se podía prever una tendencia hacia una cuota aún menor para el futuro), mayor coste de los dispositivos sobre los que se puede ejecutar la aplicación y coste para el desarrollador, ya que es necesario contar con un Mac, la última versión del SO, un iPhone, iPad o iPod, y sacarse una licencia de desarrollador de aplicaciones iOS que tiene un coste de \$99/año [5].



Análisis, diseño e implementación de un sistema para ayudar en el control de inventario. Capítulo II: Estado de la cuestión

En el caso de Android, tiene la mayor cuota de mercado y está basado en Java, lenguaje de programación que se ha utilizado mucho durante la carrera, a diferencia de iOS, cuyas aplicaciones se programan en Objective C, lenguaje que no he utilizado nunca y que hubiera tenido un periodo inicial de aprendizaje adicional.

A continuación se puede observar una tabla resumen con todos los sistemas estudiados y sus características:

Característica Sistema	1	2
Android	A+	M
iOS	M	A+
Windows Phone	B	M
Blackberry	B-	M
Symbian	B-	B
Bada	B-	B

Tabla 5 - Tabla comparativa de sistemas

1- Cuota de mercado:

- A: Alta.
- M: Media.
- B: Baja.

2- Coste de dispositivo:

- A: Alto.
- M: Medio.
- B: Bajo.

Seguidamente se explican los valores asignados en la anterior tabla a cada una de las características para cada uno de los sistemas.

Cuota de mercado:

Como se puede observar en la Tabla 4, Android tiene con diferencia la mayor cuota de mercado, por lo que se le ha asignado el valor muy alta. iOS es la segunda opción, con una diferencia destacada de la siguiente, por lo que se le ha asignado el valor media. En el caso de Windows Phone el valor es muy pequeño, por lo que se le ha asignado el valor baja. Para el resto de sistemas, se ha asignado el valor muy baja, ya que en su conjunto no llegan siquiera a la cuota de Windows Phone.



Coste de dispositivo:

- Android: en este caso existen multitud de fabricantes que sacan sus teléfonos con este sistema, por lo que hay una amplia gama donde elegir en todos los segmentos, situándose la mayoría en la gama media, por lo que el valor que se le ha asignado ha sido medio.
- iOS: en este caso, al tratarse de un sistema exclusivo de un único fabricante, la gama de productos es muy inferior, y sus costes son únicos independientemente de la tienda. Además, son terminales de gama alta, por lo que el valor que se le ha asignado es alto.
- Windows Phone: en este caso ocurre lo mismo que en el caso de Android, y por el mismo motivo se le ha asignado el valor medio.
- Blackberry: en este caso ocurre como con iOS, es decir, se trata de un único fabricante. Sin embargo, la gama de productos es mucho más amplia y estos se sitúan en todos los segmentos, por lo que el valor que se le ha asignado ha sido medio.
- Symbian: en este caso, la gama de productos es bastante amplia, pero en su mayoría se trata de dispositivos ya descatalogados y/o de gama media-baja. Por lo tanto, el valor que se le ha asignado ha sido bajo.
- Bada: en este caso ocurre lo mismo que en el caso anterior, por lo que se le ha asignado el mismo valor.

Tras haber decidido realizar la aplicación para el sistema Android, analicé qué opciones de herramientas de desarrollo podía utilizar. Hay dos tipos de herramientas: las que generan una aplicación nativa y las que generan una aplicación no nativa. Un ejemplo de la segunda opción es Phonegap.

Phonegap es un *framework* para el desarrollo de aplicaciones móviles, que permite a los programadores desarrollarlas utilizando herramientas genéricas como HTML5, CSS3 y Javascript. En las aplicaciones de Phonegap el *rendering* se realiza mediante vistas web, no con interfaces específicas de cada sistema. Los elementos HTML5, CSS3 y Javascript son empaquetados (junto con un navegador web que se encarga del *rendering*) para poder ser desplegadas en el dispositivo como una aplicación más.

A continuación se puede observar una comparativa entre Android y Phonegap. Los demás lenguajes de programación utilizados en los sistemas vistos en la Tabla 5 se han incluido también en la comparativa, para poder ver que los motivos vistos anteriormente no han sido los únicos con peso en la toma de la decisión:



Característica \ Lenguaje	1	2	3
Android (Java)	<input checked="" type="checkbox"/>	B	B+
Phonegap (HTML5, CSS3, Javascript)	<input checked="" type="checkbox"/>	B	A
iOS (Objective C)	<input checked="" type="checkbox"/>	A	B
Windows Phone (C#)	<input checked="" type="checkbox"/>	M	B
Blackberry (Java, C, C++, Qt)	<input checked="" type="checkbox"/>	A	B
Symbian (C++)	<input checked="" type="checkbox"/>	A	B
Bada (C/C++)	<input checked="" type="checkbox"/>	A-	B

Tabla 6 - Tabla comparativa de lenguajes

1- Multiplataforma.

2- Periodo de aprendizaje:

- A: Alto.
- M: Medio.
- B: Bajo.

3- Portabilidad:

- A: Alta.
- M: Media.
- B: Baja.

Seguidamente se explican los valores asignados en la anterior tabla a cada una de las características para cada uno de los lenguajes.

Multiplataforma:

En este caso Phonegap, aunque no se trata de un lenguaje propiamente dicho, es el único que se puede decir que es multiplataforma, ya que el resto está orientado a un tipo de sistema concreto, mientras que Phonegap puede ser ejecutado en todos los sistemas.



Periodo de aprendizaje:

- Android: en este caso el lenguaje para desarrollar aplicaciones para este sistema es Java, que ha sido utilizado mucho a lo largo de la carrera. Por lo tanto el valor que se le ha asignado es bajo, ya que el aprendizaje debería basarse en entender la estructura de una aplicación Android más que en entender el lenguaje en sí.
- Phonegap: en este caso el desarrollo de aplicaciones se basa en HTML5, CSS3 y Javascript, lenguajes que han sido utilizados bastante a lo largo de la carrera, motivo por el cual se le ha asignado un valor de bajo. Al igual que en el caso de Android, el aprendizaje debería basarse más en entender la estructura de una aplicación Phonegap más que en entender el lenguaje en sí (aunque Phonegap no es un lenguaje si no un *framework*).
- iOS: en este caso el lenguaje utilizado para desarrollar las aplicaciones es Objective C, lenguaje del que no se ha visto absolutamente nada en toda la carrera, y del que habría que empezar prácticamente desde el principio a aprender. Por lo tanto el valor que se le ha asignado es alto.
- Windows Phone: en este caso el lenguaje utilizado para desarrollar las aplicaciones es C#, lenguaje que ha sido utilizado durante el último año de carrera en un par de asignaturas. Por lo tanto el valor que se le ha asignado es medio.
- Blackberry: soporta Java, aunque el SDK nativo es C/C++ Qt. Ya que se tiene una base de programación en C, se le ha asignado el valor alto-medio.
- Symbian: en este caso el lenguaje nativo para Symbian es C++, aunque no es una implementación estándar. Existen múltiples SDKs para el desarrollo de aplicaciones, siendo los principales UIQ y S60. Algunos fabricantes ofrecen SDKs propios, por lo que el valor que se le ha asignado es un poco menor a alto, ya que aunque no ha sido utilizado durante la carrera, se tiene una base de programación en C.
- Bada: en este caso el lenguaje utilizado para el desarrollo de aplicaciones es un híbrido entre C y C++. Se le ha asignado el mismo valor que en el caso anterior por el mismo motivo.

Portabilidad:

En este caso, todos los lenguajes tienen asignado una portabilidad baja, ya que están focalizados hacia un único sistema. Hay dos excepciones:

- Android: portabilidad baja-media. Se ha decidido darle un pequeño valor superior al resto dada la gran cuota de mercado con la que cuenta, ya que aunque está destinado únicamente a dispositivos Android, hay una gran variedad donde elegir.
- Phonegap: portabilidad alta, ya que puede funcionar bajo cualquier sistema de los anteriormente mencionados.



Los motivos que me llevaron a elegir finalmente Phonegap son los siguientes:

- Periodo de aprendizaje muy bajo: ya había trabajado con HTML5, CSS3 y Javascript en el pasado, por lo que el aprendizaje se basaría en entender cómo funcionan las librerías de Phonegap, jQuery , jQuery Mobile y la configuración de Eclipse para poder desarrollar la aplicación correctamente, compilarla, etc.
- Alto grado de portabilidad: al estar desarrollada en Phonegap, la aplicación puede ser utilizada en Android, iOS, Windows Phone, Blackberry OS, Bada, Symbian, etc. lo que permite no restringir a los usuarios al uso de un sistema concreto.

Por el contrario, las aplicaciones desarrolladas en Phonegap tienen un inconveniente, y es que no podemos exprimir al máximo la potencia de cada SO al no tratarse de aplicaciones nativas, obteniendo un nivel algo inferior de eficiencia.

2.4.2 jQuery

Se decidió utilizar una librería Javascript con el fin de simplificar la manera de interactuar con los archivos HTML, manejar eventos, etc. permitiendo realizar funcionalidades complejas con pocas líneas de código, ahorrando esfuerzo y tiempo.

Se investigó acerca de los diferentes *frameworks* disponibles para la realización de estas labores, siendo los más destacados los siguientes:

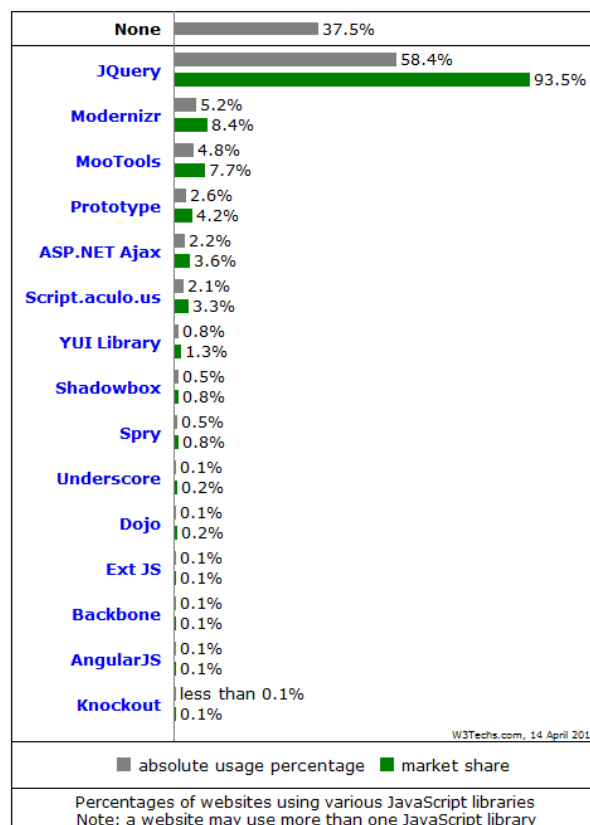


Ilustración 9 - Comparación de uso entre frameworks



De la anterior lista [6], se barajaron dos opciones: Dojo [7] y jQuery [8]. El motivo de escoger dichas opciones, es que durante la carrera se han visto, aunque sea a modo introductorio.

En el caso de Dojo se trata de un *framework* que contiene APIs y *widgets* para facilitar el desarrollo de aplicaciones web que utilicen tecnología AJAX. Fue visto durante la carrera en la asignatura Interfaces de Usuario, por lo que se conocía su uso, pero de un modo muy superficial.

En el caso de jQuery, éste ofrece un gran potencial, y al igual que Dojo, había sido utilizado alguna vez durante la carrera en diferentes asignaturas. El motivo que realmente tuvo un gran peso en la toma de la decisión del *framework* a utilizar fue que el uso de jQuery está mucho más extendido que el de otros *framework* similares (ver Ilustración 9), por lo que el soporte que podemos encontrar es mayor.

A continuación se puede observar una comparativa entre los distintos *framework*:

Característica Framework	1	2	3
	Dojo	jQuery	
	M-	A	B-
	B+	A	A+

Tabla 7 - Tabla comparativa de frameworks

1 – Periodo de aprendizaje:

- A: Alto.
- M: Medio.
- B: Bajo.

2 – Potencial:

- A: Alto.
- M: Medio.
- B: Bajo.

3- Soporte:

- A: Alto.
- M: Medio.
- B: Bajo.

Seguidamente se explican los valores asignados en la anterior tabla a cada una de las características para cada uno de los *framework*.



Periodo de aprendizaje:

- Dojo: en este caso fue visto (aunque muy superficialmente) durante la carrera, por lo que el valor que se le ha asignado es medio-bajo, ya que se tiene una mínima base.
- jQuery: en este caso pasa lo mismo que en el anterior: se ha utilizado en diferentes momentos de la carrera, por lo que el valor que se le ha asignado es bajo-medio. El motivo de que el valor sea mayor que para Dojo es que se ha utilizado más.

Potencial:

En ambos casos al potencial se le ha asignado un valor de alto, ya que permiten hacer cosas similares.

Soporte:

- Dojo: en este caso se le ha asignado el valor muy bajo, dado que el porcentaje de uso de este *framework* es ínfimo.
- jQuery: en este caso se le ha asignado el valor muy alto, ya que tiene un alto porcentaje de uso y prácticamente la totalidad de la cuota de mercado.

Para estimar la facilidad de encontrar soporte, se han realizado búsquedas en Google con el nombre del *framework* + *problems*. Los resultados pueden verse en la siguiente tabla:

Framework	Número de resultados
Dojo	121.000 resultados
jQuery	36.100.000 resultados

Tabla 8 - Tabla comparativa de soporte estimado frameworks

Como se puede observar el soporte estimado para jQuery es del orden de casi 300 veces más al soporte para Dojo.

2.4.3 jQuery Mobile

Para la realización de la interfaz de la aplicación se barajaron dos *framework*: Sencha Touch [9] y jQuery Mobile [10].

Ambos se tratan de librerías HTML5 y Javascript que facilitan la creación de aplicaciones para dispositivos móviles.

jQuery Mobile ofrece un gran potencial, aunque Sencha Touch puede considerarse al mismo nivel. El motivo que decantó la balanza a favor de jQuery Mobile fue que sería más fácil integrarlo, ya que se trata de un *plugin* de jQuery, librería que iba a ser utilizada para



la manipulación de los HTML, manejo de eventos, etc. como se ha podido ver en el anterior apartado.

Como puntos a destacar:

- Es compatible con la mayoría de plataformas móviles y navegadores web de escritorio.
- La curva de aprendizaje es mínima porque la sintaxis es muy similar a jQuery.
- Permite cambiar fácilmente el tema de la aplicación e incluso, crear temas propios.
- Se auto optimiza para los diferentes tamaños de pantalla.

A continuación se puede observar una comparativa entre los distintos *framework*:

Característica \ Framework	1	2	3
Sencha Touch	B	A	M-
jQuery Mobile	B	A	A

Tabla 9 - Tabla comparativa de frameworks para la interfaz

1 – Periodo de aprendizaje:

- A: Alto.
- M: Medio.
- B: Bajo.

2 – Potencial:

- A: Alto.
- M: Medio.
- B: Bajo.

3- Integración:

- A: Alta.
- M: Media.
- B: Baja.

Seguidamente se explican los valores asignados en la anterior tabla a cada una de las características para cada uno de los *framework*.

Periodo de aprendizaje:

En ambos casos el periodo de aprendizaje se ha considerado como bajo, ya que ambas se tratan de librerías Javascript, y dicho periodo sería similar indistintamente de la opción elegida.



Potencial:

Del mismo modo, el potencial de ambas alternativas se ha considerado como alto, ya que ambas son muy similares en cuanto a lo que ofrecen.

Integración:

Este es el punto crítico que ha decantado la elección a favor de jQuery Mobile. En el caso de Sencha Touch, el valor que se le ha asignado ha sido medio-bajo, ya que se trata de una librería totalmente independiente de jQuery. Sin embargo, jQuery Mobile se trata de un *plugin* de jQuery, que como se ha visto en el anterior apartado ha sido el *framework* seleccionado, por lo que la integración es mucho mayor, y se le ha asignado el valor alto.

2.4.4 Web SQL

Para soportar los datos de la aplicación se estuvieron barajando diferentes posibilidades. Dado el carácter que se le quería dar a la solución, se pensó en una opción que permitiera sincronización entre diferentes dispositivos de una manera sencilla.

Mirando posibles soluciones que permitieran agregar esta característica, se descubrió la existencia de los *datastores* de *Dropbox Sync* [11]. Los *datastores* son una forma sencilla de guardar los datos de la aplicación en la nube, permitiendo sincronizarlos en diferentes dispositivos. Los *datastores* contienen un conjunto de tablas, y cada tabla es una colección de registros. Cada registro consiste en un conjunto de campos, cada uno con un nombre y un valor. Cada registro tiene un ID y puede tener un número cualquiera de campos. En diferencia a SQL, las tablas en los *datastores* no siguen un esquema, por lo que cada registro tiene un número arbitrario de campos. Se probó a utilizar este elemento para soportar la base de datos de la aplicación, pero no se consiguió que funcionara, ni siquiera el ejemplo que se puede encontrar en la API de *Dropbox*. Tras contactar con ellos, se descartó la idea de utilizarlo, ya que decían que no sabían cuál podía ser el problema, y que quizás no era compatible su ejecución en dispositivos móviles utilizando Javascript.

Tras descartar esta opción, se barajaron otras dos. La primera de ellas: *Local Storage* [12]. Este tipo de almacenamiento proporciona almacenamiento síncrono de pares de clave-valor. Destaca por su sencillez, pero finalmente la opción elegida la segunda: *Web SQL Database* [13].

La especificación de *Web SQL Database* ofrece unas tablas de bases de datos con mayor funcionalidad, a las que se accede a través de consultas SQL. Dado que ya había trabajado con SQL y estaba familiarizado con sus sentencias, se decidió que era la opción más conveniente.

A continuación se puede observar una comparativa entre los distintos sistemas de almacenamiento de datos:



Característica \ Sistema	1	2	3
Dropbox Sync: Datastores	M+	A+	A*
Local Storage	M-	M	B
Web SQL	B	A	B

Tabla 10 - Tabla comparativa de opciones de almacenamiento

1 – Periodo de aprendizaje:

- A: Alto.
- M: Medio.
- B: Bajo.

2 – Potencial:

- A: Alto.
- M: Medio.
- B: Bajo.

3 – Sincronización:

- A: Alta.
- M: Media.
- B: Baja.

Seguidamente se explican los valores asignados en la anterior tabla a cada una de las características para cada uno de los elementos.

Periodo de aprendizaje:

- Dropbox Sync: en este caso se trata de un sistema relativamente nuevo, por lo que el aprendizaje es casi desde cero. Por ello el valor que se le ha asignado es de Medio Alto.
- Local Storage: al igual que en el caso anterior, el aprendizaje es casi desde cero ya que nunca he utilizado esta tecnología, pero el manejo es más sencillo que en el caso de los *datastores* de Dropbox. Por ello el valor que se le ha asignado es de Medio Bajo.
- Web SQL: en este caso para utilizar Web SQL se utilizan sentencias SQL, lenguaje que ha sido aprendido durante la carrera, y que, por tanto, requeriría un periodo de aprendizaje prácticamente nulo, ya que se trata de refrescar conocimientos. Por ello el valor que se le ha asignado es de Bajo.



Potencial:

- Dropbox Sync: en el caso de Dropbox Sync se trata de una opción con muchísimo potencial, ya que permite flexibilidad en las tablas (cada registro puede tener un determinado número de columnas) y sincronización entre dispositivos, lo que sería, sin duda alguna, una característica muy interesante que ofrecer. Por ello el valor que se le ha asignado ha sido Muy Alto.
- Local Storage: en este caso el potencial que nos ofrece no es muy grande, permitiendo guardar pares de clave valor, pero sin ir más allá. Por ello el valor que se le ha asignado ha sido Medio.
- Web SQL: en este caso el potencial que nos ofrece Web SQL es mayor en cuanto a gestión de los datos, pero menor si lo comparamos con Dropbox Sync en el sentido de que no permite sincronización automática entre dispositivos. Por ello el valor que se le ha asignado ha sido un poco menor, Alto.

Sincronización:

En los casos de Local Storage y Web SQL se le ha asignado el valor baja, ya que hay que programar una sincronización propia. En el caso de Dropbox Sync debería tener el valor alta, ya que permite realizar la sincronización automáticamente, almacenando la base de datos en la nube. Sin embargo, y dado que no se ha conseguido hacer funcionar, el valor asignado no se tiene en cuenta en la comparativa.

2.4.5 Plugin LazarSoft QR

Se quería incluir en la aplicación la opción de poder decodificar códigos QR, de modo que se pudiera agilizar aún más el control del inventario. Para ello se estuvieron barajando dos opciones: recurrir a aplicaciones de terceros para realizar la decodificación, o bien, crear dicha funcionalidad dentro de la propia aplicación desarrollada.

A continuación se puede observar una comparativa entre los distintos sistemas de decodificación de códigos QR:

Característica \ Sistema	1	2	3	4	5
	1	2	3	4	5
Aplicación de terceros	M-	M	M	B	M
Funcionalidad interna	M-	A	A	A	B

Tabla 11 - Tabla comparativa de opciones de decodificación de códigos QR

1 – Periodo de aprendizaje:

- A: Alto.
- M: Medio.
- B: Bajo.



2 – Profesionalidad:

- A: Alta.
- M: Media.
- B: Baja.

3– Fiabilidad:

- A: Alta.
- M: Media.
- B: Baja.

4 – Comodidad:

- A: Alta.
- M: Media.
- B: Baja.

5 – Tiempo de respuesta:

- A: Alto.
- M: Medio.
- B: Bajo.

Seguidamente se explican los valores asignados en la anterior tabla a cada una de las características para cada uno de los sistemas.

Periodo de aprendizaje:

En este caso, ambos sistemas requerirían un periodo de aprendizaje similar, ya que en ambos casos se trataría de llamadas a funciones de librería que nunca se han utilizado, por lo que el valor asignado ha sido medio-bajo.

Profesionalidad:

- Aplicación de terceros: en este caso, el tener que abrir otra *app* ajena a iNventariator para hacer la decodificación del código QR, dota a la aplicación desarrollada de cierta falta de profesionalidad de cara al usuario, por lo que el valor que se le ha asignado es media.
- Funcionalidad interna: por el contrario, lo deseable sería que dicha funcionalidad fuera interna, es decir, que tanto la captura como la decodificación del código QR se hiciera de forma interna desde la misma aplicación desarrollada, lo que dota a la aplicación desarrollada de una imagen mucho más profesional. Por lo tanto el valor que se le ha asignado es alta.

Fiabilidad:

- Aplicación de terceros: en este caso, y aunque la aplicación externa tenga buena fiabilidad, se depende de un tercero para la solución de problemas en caso de encontrarlos, y no depende de uno mismo el correcto funcionamiento de la misma. Por tanto, el valor que se le ha asignado es media.



- Funcionalidad interna: en este caso, al haber desarrollado la aplicación en la propia aplicación, es más sencillo corregir los posibles problemas encontrados, y no depende del correcto funcionamiento de una aplicación externa para funcionar, por lo que se le ha asignado el valor alta.

Comodidad:

- Aplicación de terceros: en este caso se “obliga” al usuario a descargar e instalar una aplicación distinta para poder utilizar la funcionalidad, lo que puede provocar cierta incomodidad. Por tanto, el valor que se le ha asignado es baja.
- Funcionalidad interna: en este caso el usuario no tiene que preocuparse de instalar ninguna aplicación adicional, sólo de utilizar la funcionalidad, por lo que el valor que se le ha asignado es alta.

Tiempo de respuesta:

- Aplicación de terceros: en este caso, al utilizar la funcionalidad ha de abrirse una aplicación distinta, procesar los datos, y devolver la información a la aplicación original, lo que conlleva cierto retraso en el tiempo de respuesta. Por ello el valor que se le ha asignado es medio.
- Funcionalidad interna: en este caso la funcionalidad se hace internamente, sin necesidad de abrir nuevas aplicaciones, por lo que el valor que se le ha asignado es bajo.

Por todo lo visto anteriormente, finalmente la opción escogida fue desarrollar dicha funcionalidad dentro de la propia aplicación.

Para su implementación se probó el uso de diferentes *plugins*, como por ejemplo, los *plugins* de Jerome Etienne [14] y Marco Del Corno [15].

Con ninguno de ellos se consiguió la funcionalidad requerida. Finalmente el *plugin* que se utilizó fue el LazarSoft QR [16].

Se trata de un *plugin* desarrollado en Javascript que permite la decodificación de códigos QR, y que es un *port* de *ZXing QRCode Scanner* [17], que está desarrollado en Java. En este apartado no se encontraron otras alternativas reales, ya que todas las que se probaron (incluido el citado *plugin* de *LazarSoft*), fallaban al intentar decodificar la imagen que contenía el código QR. Finalmente, y tras solventar los problemas que ocurrían con el *plugin LazarSoft*, se adoptó como sistema para la decodificación de códigos QR. Al no haber otras alternativas funcionales, no se ha podido comparar las características de las diferentes alternativas.

2.4.6 Cifrado

Para el cifrado de los datos se estuvieron barajando varios algoritmos, entre los que cabe destacar MD5 y SHA1 [18]. Ambos se tratan de las conocidas como función *hash* o funciones resumen.



Una función resumen produce una salida de un tamaño predeterminado (dependiendo del tipo de algoritmo) para una entrada dada, es decir, el tamaño de la función resumen será el mismo para una palabra que para un texto completo.

Con MD5, para un ataque basado en colisiones [19] sería necesario realizar 2^{64} operaciones. En el caso de SHA1, sería necesario realizar 2^{80} operaciones. Sin embargo, en 2004 se encontró una debilidad matemática [20] que permitiría encontrar colisiones de *hash* más rápido, reduciendo la complejidad de 2^{80} a 2^{69} , lo que sigue siendo computacionalmente inviable. El motivo por el que finalmente se ha elegido SHA1 es por ese extra de complejidad que ofrece frente a MD5, y porque los datos que se desean proteger no requieren algoritmos más complejos como SHA256 o SHA512, que producen salidas de mayor longitud.

A continuación se puede observar una comparativa entre los distintos algoritmos de cifrado:

Característica \ Algoritmo	1	2	3
	1	2	3
MD5	B	A	M
SHA1	B	A	M+

Tabla 12 - Tabla comparativa de algoritmos de cifrado

1 – Periodo de aprendizaje:

- A: Alto.
- M: Medio.
- B: Bajo.

2 – Potencial:

- A: Alto.
- M: Medio.
- B: Bajo.

3 – Resistencia a colisiones:

- A: Alta.
- M: Media.
- B: Baja.

Seguidamente se explican los valores asignados en la anterior tabla a cada una de las características para cada uno de los algoritmos.



Periodo de aprendizaje:

En ambos casos se trata de librerías Javascript, en las que se llama a la función *hash* pasándola por parámetro el valor que se desea cifrar, y devuelve su función resumen, por lo que el periodo de aprendizaje es mínimo.

Potencial:

En ambos casos el potencial es alto, ya que se trata de funciones irreversibles, es decir, una vez cifrado el texto no es posible volver al texto original.

Resistencia a colisiones:

- MD5: en este caso, como se ha comentado anteriormente, un ataque basado en colisiones necesitaría 2^{64} operaciones, que se encuentra dentro de los límites de las capacidades actuales de cálculos, y probablemente con el paso del tiempo se pueda romper esta función de forma trivial. Por ello el valor que se le ha asignado es medio.
- SHA1: en el caso de SHA1, un ataque basado en colisiones necesitaría de 2^{80} operaciones, si bien, como se ha comentado antes, en 2004 consiguieron reducir la complejidad a 2^{69} operaciones. Por ello el valor que se le ha asignado es medio-alto.

Para saber cómo se ha utilizado dicho algoritmo, se puede consultar el apartado de Implementación.



3 Análisis, diseño, implementación e implantación

3.1 Introducción

En este capítulo se expondrán los aspectos asociados al desarrollo del proyecto propiamente dicho. Se comentará y explicará brevemente la metodología que ha sido utilizada, el ciclo de vida y el marco regulador. Se realizará el análisis del sistema, incluyendo los casos de uso y los requisitos que ha de satisfacer el mismo. Se mostrará el diseño del sistema, incluyendo los diferentes diagramas de navegación de la aplicación, el esquema de la base de datos y el diseño de las pruebas de aceptación que ha de pasar el producto para cumplir con los requerimientos. Se explicarán aquellos detalles de la implementación que más ha costado realizar y por último, se explicará cómo llevar a cabo la implantación del sistema.

3.1.1 Metodología utilizada

En esta sección se explicará la metodología que ha sido aplicada para la realización del Trabajo de Fin de Grado. Dado que el objetivo del mismo es la creación de un prototipo de aplicación que cumpla con la funcionalidad básica del sistema analizado, se ha decidido aplicar una metodología adaptada. Durante la carrera se han visto dos tipos de metodologías distintas en diferentes asignaturas: las metodologías clásicas o pesadas, concretamente Métrica V3, en DPDS; y las metodologías ágiles, concretamente *Scrum*, en MDV, TADS y TIW.

A continuación se explicarán brevemente las características de dichas metodologías.

3.1.1.1 Metodologías Pesadas: Métrica V3

Métrica V3 es una metodología de planificación, desarrollo y mantenimiento de sistemas de información, y está promovida por el Ministerio de Hacienda y Administraciones Públicas del gobierno de España [21]. Tiene como objetivo la sistematización de actividades del ciclo de vida de los proyectos *software* en el ámbito de las administraciones públicas. Esta metodología está basada en el modelo de procesos del ciclo de vida de desarrollo ISO/IEC 12207, y proporciona un marco de procedimientos a seguir, técnicas a utilizar, fases del ciclo de vida, etc. Se engloba dentro de las Metodologías Pesadas, que están orientadas hacia el proceso del *software*. Dicho proceso es rígido, basado en numerosas políticas o normas provenientes de estándares seguidos por el entorno de desarrollo, y no existen puntos de control durante el mismo, lo que provoca que la calidad del *software* entregado sea detectada al final, cuando ya es demasiado costoso de mejorar.



Análisis, diseño e implementación de un sistema para ayudar en el control de inventario.
Capítulo III: Análisis, diseño, implementación e implantación.

Se caracteriza por una documentación extensa, la cual trata de garantizar la calidad del producto en un tiempo y costo determinado.

3.1.1.2 Metodologías Ágiles: Scrum

Scrum es una metodología de trabajo para la gestión y desarrollo de *software*, que está basada en un proceso iterativo e incremental. Éste es un proceso básico, a diferencia de la rigidez que caracteriza a Métrica V3. En este proceso se definen puntos de control para mejorar el mismo, es lo que se conoce como retrospectivas.

Está orientada hacia las necesidades del cliente, de forma que está especialmente preparada para cambios durante el desarrollo del proyecto.

Se caracteriza por entregar al cliente *software* funcional en cada iteración, lo que permite detectar la calidad del mismo, y mejorando con un coste bajo aquellos aspectos que sea necesario corregir. La documentación en este caso es la mínima para permitir la correcta gestión del proyecto.

Generalmente, las metodologías ágiles son procesos iterativos en los que se entrelazan la especificación, el diseño, el desarrollo y las pruebas. El *software* se va desarrollando a través de incrementos, en donde en cada incremento se incluyen nuevas funcionalidades al sistema.

Con estas metodologías el cliente se compromete más con el proyecto, ya que lo ve crecer en cada iteración. Además, le permite hacer modificaciones funcionales o de prioridad de los requisitos del *software* al inicio de cada iteración con el fin de enfocarse mejor hacia los objetivos del negocio.

En conclusión, comparando las metodologías ágiles con las pesadas:

- Los individuos y sus iteraciones son más importantes que los procesos y herramientas.
- *Software* funcionando es más importante que una documentación exhaustiva.
- La colaboración con los clientes es más importante que la negociación de los contratos.
- La respuesta ante los cambios es más importante que el seguimiento de un plan.

3.1.2 Ciclo de vida

El ciclo de vida que se ha seguido es un ciclo de vida adaptado, basado en características de las metodologías pesadas y de las metodologías ágiles. A continuación, se puede observar un diagrama que resume las fases por las que se ha ido pasando:

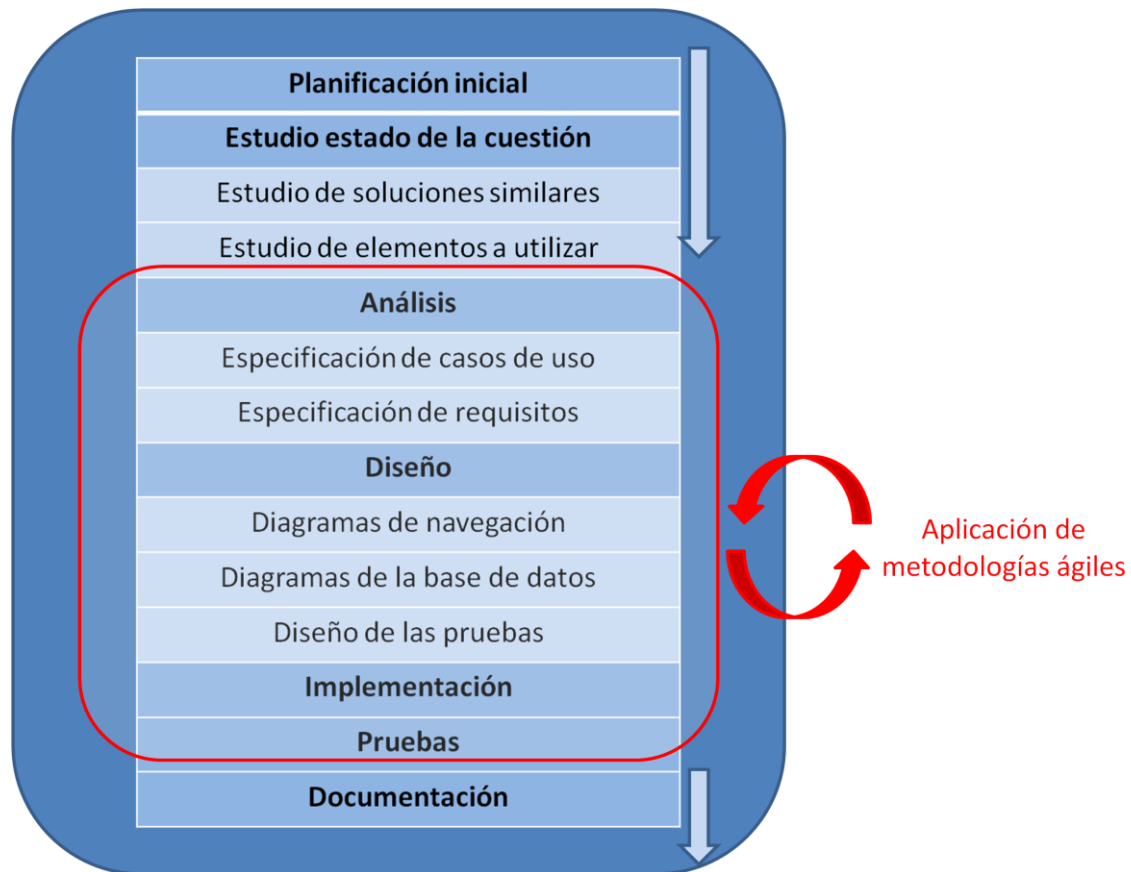


Ilustración 10 - Ciclo de vida

Tras el inicio del proyecto y desarrollándolo con una metodología basada en metodologías pesadas, me di cuenta al llegar a la fase de análisis de que para el desarrollo de la aplicación en sí, esta metodología no era productiva. Los motivos por los que para las fases de desarrollo de la aplicación se haya decidido aplicar una metodología basada en metodologías ágiles son los siguientes:

- Se trata de un entorno en el que se ha ido aprendiendo a la vez que se iba desarrollando, y en una plataforma bastante nueva, sujeta a muchos cambios y actualizaciones, por lo que su desarrollo iba a ser igual de variable.
- Durante las primeras fases del ciclo de vida del desarrollo, hay que tomar decisiones importantes, pero en estas etapas tempranas se dispone de poca información, hay gran incertidumbre y por lo tanto, una alta probabilidad de cometer errores. Dichos errores no se detectan hasta el final, cuando su resolución es muy costosa.
- No se dispone de experiencia previa en el desarrollo de proyectos, por lo que no se puede aprovechar todo el potencial que puede dar de sí una metodología pesada.

Las fases incluidas en la “aplicación de metodologías ágiles”, se desarrollarán mediante *sprints*. Un *sprint* es el período en el cual se lleva a cabo el trabajo en sí. Su duración es establecida por el equipo al inicio del proyecto y debe ser constante. En este caso su duración será de una semana. Al finalizar cada *sprint*, el equipo debe presentar los avances logrados, y el *software* obtenido es un producto funcional que se entrega al cliente. Durante el *sprint* no se pueden modificar los objetivos ni los requisitos.



Análisis, diseño e implementación de un sistema para ayudar en el control de inventario.
Capítulo III: Análisis, diseño, implementación e implantación.

Por lo tanto, el ciclo de vida del proyecto sería un ciclo en cascada, en el que el conjunto de fases englobadas dentro de “aplicación de metodologías ágiles” se realizará de un modo iterativo e incremental. Cada *sprint* queda dividido a su vez en cuatro fases: planificación del *sprint*, seguimiento del *sprint*, revisión del *sprint* y retrospectiva del *sprint*.

3.1.2.1 Planificación del sprint

Es la reunión que se tiene al inicio de cada *sprint*. En esta fase se define una lista de requisitos del sistema. En cada iteración, esta lista es revisada. También se seleccionan los objetivos, se identifica y comunica el trabajo que es probable que se realice en esa iteración, y el tiempo que tomará hacer dicho trabajo. Tiene ocho horas como límite de duración.

3.1.2.2 Seguimiento del sprint

En esta fase tienen lugar los *Daily Scrum* o *Stand-up meeting*. Se tratan de reuniones diarias en las que se comprueba el avance de las tareas que se tienen que realizar durante el *sprint*. En dichas reuniones están presentes el *Scrum Master* y el equipo de desarrollo. Estas reuniones deben tener lugar siempre en la misma ubicación y a la misma hora todos los días, y tienen una duración fija de 15 minutos. En ellas cada miembro del equipo ha de contestar tres preguntas:

- ¿Qué has hecho desde ayer?
- ¿Qué es lo que harás hasta la reunión de mañana?
- ¿Has tenido algún problema que te haya impedido alcanzar tu objetivo?

3.1.2.3 Revisión del sprint

Es la reunión que se tiene tras la finalización de cada *sprint* con el fin de analizar y revisar el mismo. En ella se revisa el trabajo que ha sido completado y el que no ha podido ser completado. Se presenta el trabajo completado a los interesados. Esta demo es importante ya que mejora el *feedback* de los interesados, hay un reconocimiento del trabajo hecho, y brinda la posibilidad de corregir los puntos más flojos para el siguiente *sprint*. El trabajo incompleto no puede ser demostrado. Tiene cuatro horas como límite de duración.

3.1.2.4 Retrospectiva del sprint

Una vez superado cada *sprint*, se lleva a cabo una retrospectiva del *sprint*, en la que todos los miembros del equipo comentan sus impresiones sobre el *sprint* recién superado. El propósito de la misma es la mejora continua del proceso. Tiene un tiempo fijo de cuatro horas de duración.



3.1.3 Roles

Para el correcto desarrollo de *Scrum* es necesario adquirir una serie de roles a la hora de realizar el proyecto:

3.1.3.1 Product Owner o propietario del producto

Representa la voz del cliente. Conoce el entorno de negocio del cliente y es el responsable de que el equipo *Scrum* trabaje de forma adecuada desde la perspectiva del negocio. El *Product Owner* escribe los requisitos y los prioriza. Además toma decisiones que afectan al resultado final, elige fechas de lanzamiento y el retorno de inversión.

3.1.3.2 Scrum Master o facilitador

Su trabajo consiste en eliminar los obstáculos que impiden que el equipo alcance el objetivo del *sprint*. En esta figura recae toda la responsabilidad de funcionamiento del modelo. No se trata de un líder de equipo, ya que los equipos son auto organizados, si no que actúa como una protección entre el equipo y cualquier influencia que le distraiga. Se asegura de que el proceso se utiliza como es debido y hace que las reglas se cumplan. Interactúa con el equipo de desarrollo, el cliente y los gestores.

3.1.3.3 Equipo de desarrollo

Este equipo debe contar con personas con las habilidades necesarias para realizar el trabajo de análisis, diseño, pruebas, etc. Tiene la autoridad para decidir las acciones necesarias y auto organizarse con el fin de alcanzar los objetivos del *sprint*. Tiene la responsabilidad de entregar el producto

3.1.3.4 Stakeholders

Es el conjunto de gente que hace posible el proyecto (clientes, proveedores, vendedores, etc.) y para quienes el proyecto producirá el beneficio acordado que justifica su producción. Sólo participan directamente durante las revisiones del *sprint*.



3.1.4 Marco regulador

En esta sección se recogen las normativas técnicas y legales que afectan al presente Trabajo de Fin de Grado. En este caso, la aplicación desarrollada durante el mismo utiliza una base de datos que almacena registros de carácter personal, por lo que la ley que la regula es la Ley Orgánica 15/1999, del 13 de diciembre, de Protección de Datos de Carácter Personal [22]. Para proteger el acceso a la aplicación y por tanto, los datos que almacena, se establece una pantalla de *login* en la que los usuarios tienen que autenticarse, de modo que sólo los usuarios registrados puedan acceder. Para proteger la aplicación de posibles atacantes que intenten hacerse con la contraseña de acceso, se cifrará la misma, como se explicará en el apartado Implementación.

3.2 Análisis

En este apartado se mostrarán los resultados de la etapa de análisis del sistema, donde se recogen los casos de uso y los requisitos que se extraen de los mismos. En resumen, se especificará qué debe hacer el sistema, y cómo lo debe llevar a cabo.

3.2.1 Casos de uso

En esta sección se muestra la funcionalidad del sistema a desarrollar, mediante diagramas con los diferentes casos de uso. Además, podremos encontrar una definición textual de todos los casos de uso presentes en los diagramas.

3.2.1.1 Diagramas de casos de uso

De un modo general, tenemos 3 casos de uso, como se puede apreciar en la siguiente figura.



Ilustración 11 - Diagrama de casos de uso genérico



Estos casos de uso generales pueden ser desglosados a su vez en otros, como se puede observar en los siguientes diagramas.

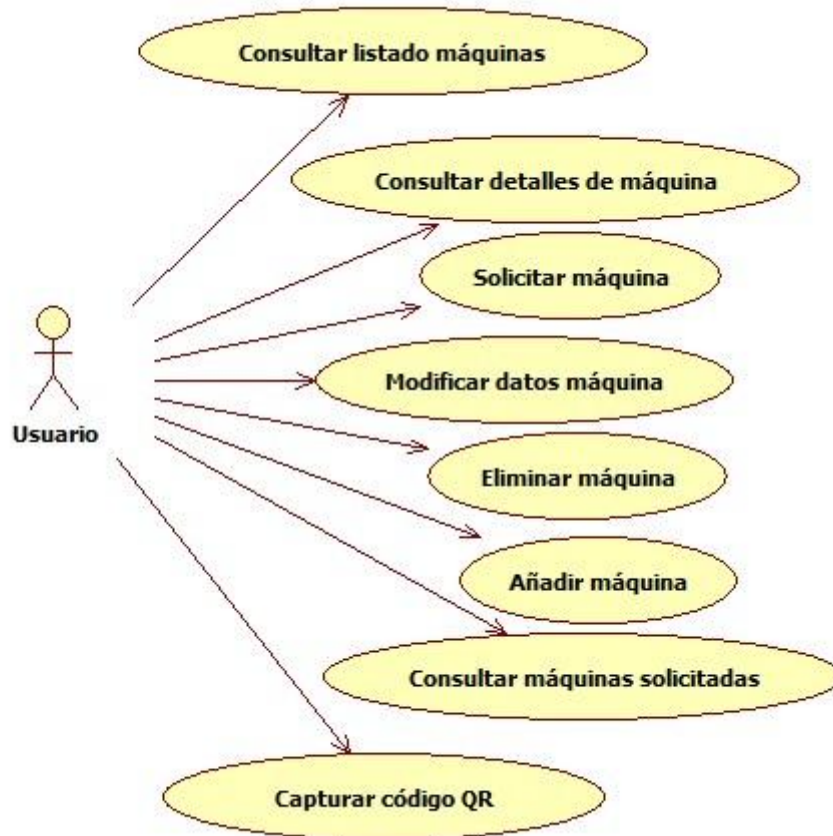


Ilustración 12 - Diagrama de casos de uso para el control de máquinas



Ilustración 13 - Diagrama de casos de uso para el control de usuarios

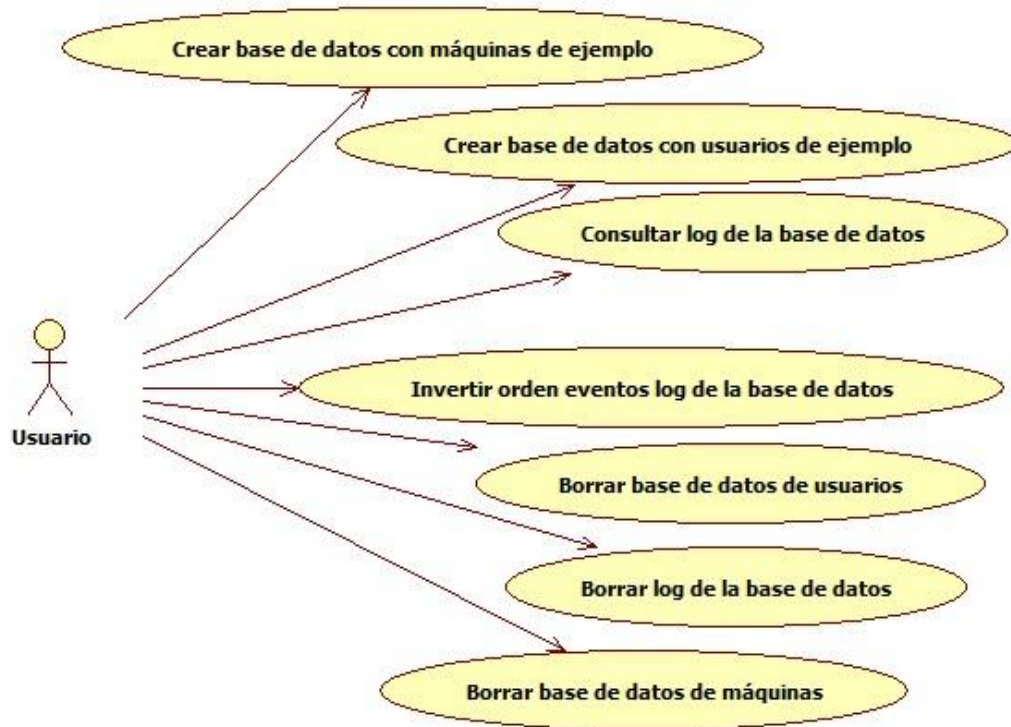


Ilustración 14 - Diagrama de casos de uso para la administración

3.2.1.2 Definición textual de los casos de uso

En esta sección se muestran de un modo textual y con un mayor nivel de detalle los casos de uso identificados en los diagramas del apartado anterior. Los atributos utilizados en los casos de uso son los siguientes:

- **Identificador:** permite identificar inequívocamente a todos y cada uno de los casos de uso a lo largo del proyecto. Se compone de la abreviatura de Caso de Uso (CU) y un número correlativo.
- **Nombre:** descripción corta que permite referirse al caso de uso de manera sencilla, y proporciona un resumen de su contenido.
- **Descripción:** contiene una breve descripción del caso de uso, resumiendo lo que puede hacer el usuario.
- **Actores:** indica que personas pueden realizar un determinado caso de uso.
- **Precondiciones:** condiciones previas que deben ser cumplidas para poder realizar una determinada operación.
- **Escenario normal:** ejecución del caso de uso paso a paso.
- **Escenario alternativo:** condiciones excepcionales que afectan al escenario, y respuestas del sistema ante estas situaciones.
- **Postcondiciones:** estado en el que queda el sistema tras realizar una determinada operación.



CU-001: Consultar listado máquinas	
Descripción	Permite al usuario examinar un listado con todas las máquinas existentes en la base de datos de la aplicación.
Actores	Usuario de la aplicación.
Precondiciones	Iniciar la aplicación, estar logueado y que existan máquinas en la base de datos.
Escenario normal	1. Pulsar en la opción máquinas del menú principal.
Escenario alternativo	No existe ninguna máquina en la base de datos: <ul style="list-style-type: none">• Se muestra un mensaje indicándolo al usuario.
Postcondiciones	El listado es mostrado por pantalla al usuario.

Tabla 13 - CU-001: Consultar listado máquinas

CU-002: Consultar detalles de máquina	
Descripción	Permite al usuario examinar los detalles de una determinada máquina.
Actores	Usuario de la aplicación.
Precondiciones	Iniciar la aplicación, estar logueado y que existan máquinas en la base de datos.
Escenario normal	1. Pulsar en la opción máquinas del menú principal. 2. Pulsar sobre el nombre de alguna máquina del listado.
Escenario alternativo	-
Postcondiciones	Los detalles de la máquina son mostrados por pantalla al usuario.

Tabla 14 - CU-002: Consultar detalles de máquina



CU-003: Solicitar máquina	
Descripción	Permite al usuario solicitar una determinada máquina.
Actores	Usuario de la aplicación.
Precondiciones	Iniciar la aplicación, estar logueado y que existan máquinas en la base de datos.
Escenario normal	<ol style="list-style-type: none">1. Pulsar en la opción máquinas del menú principal.2. Pulsar sobre el nombre de alguna máquina del listado.3. Pulsar sobre el botón solicitar en la parte inferior de los detalles de la máquina.
Escenario alternativo	La máquina que se intenta solicitar ya está solicitada por otro usuario: <ul style="list-style-type: none">• Se muestra un mensaje indicándoselo al usuario.
Postcondiciones	La máquina es solicitada.

Tabla 15 - CU-003: Solicitar máquina

CU-004: Modificar datos máquina	
Descripción	Permite al usuario modificar los datos de una determinada máquina.
Actores	Usuario de la aplicación.
Precondiciones	Iniciar la aplicación, estar logueado y que existan máquinas en la base de datos.
Escenario normal	<ol style="list-style-type: none">1. Pulsar en la opción máquinas del menú principal.2. Pulsar sobre el nombre de alguna máquina del listado.3. Pulsar sobre el botón modificar en la parte inferior de los detalles de la máquina.4. Modificar aquellos campos que se considere oportunos.5. Pulsar sobre el botón modificar.
Escenario alternativo	El formato de alguno de los campos modificados no es correcto: <ul style="list-style-type: none">• Se muestra un mensaje indicándoselo al usuario.
Postcondiciones	Los datos de la máquina son actualizados.

Tabla 16 - CU-004: Modificar datos máquina



CU-005: Eliminar máquina	
Descripción	Permite al usuario eliminar una máquina de la base de datos de la aplicación.
Actores	Usuario de la aplicación.
Precondiciones	Iniciar la aplicación, estar logueado y que existan máquinas en la base de datos.
Escenario normal	<ol style="list-style-type: none">1. Pulsar en la opción máquinas del menú principal.2. Pulsar sobre el aspa que hay a la derecha de los nombres de máquina de alguna máquina del listado.3. Pulsar sobre el botón aceptar.
Escenario alternativo	-
Postcondiciones	La máquina es eliminada de la base de datos.

Tabla 17 - CU-005: Eliminar máquina

CU-006: Añadir máquina	
Descripción	Permite al usuario añadir una máquina a la base de datos de la aplicación.
Actores	Usuario de la aplicación.
Precondiciones	Iniciar la aplicación y estar logueado.
Escenario normal	<ol style="list-style-type: none">1. Pulsar en la opción máquinas del menú principal.2. Pulsar en la opción añadir máquina del menú máquinas.3. Completar los datos del formulario.4. Pulsar sobre el botón añadir.
Escenario alternativo	El formato de alguno de los campos no es correcto: <ul style="list-style-type: none">• Se muestra un mensaje indicándoselo al usuario.
Postcondiciones	La máquina es añadida a la base de datos.

Tabla 18 - CU-006: Añadir máquina



CU-007: Consultar máquinas solicitadas	
Descripción	Permite al usuario consultar un listado de las máquinas que han sido solicitadas.
Actores	Usuario de la aplicación.
Precondiciones	Iniciar la aplicación, estar logueado y que existan máquinas en la base de datos.
Escenario normal	<ol style="list-style-type: none">1. Pulsar en la opción máquinas del menú principal.2. Pulsar en la opción solicitadas del menú máquinas.
Escenario alternativo	No existe ninguna máquina solicitada en la base de datos: <ul style="list-style-type: none">• Se muestra un mensaje indicándolo al usuario.
Postcondiciones	Se muestra un listado con las máquinas que están solicitadas.

Tabla 19 - CU-007: Consultar máquinas solicitadas

CU-008: Capturar código QR	
Descripción	Permite al usuario capturar el código QR de una máquina.
Actores	Usuario de la aplicación.
Precondiciones	Iniciar la aplicación, estar logueado y que la máquina tenga una pegatina con un código QR con su nombre o exista una foto de dicha pegatina guardada en la tarjeta de memoria.
Escenario normal	<ol style="list-style-type: none">1. Pulsar en la opción códigos QR del menú principal.2. Pulsar en la opción capturar en el menú códigos QR.3. Apuntar la cámara hacia el código QR y pulsar sobre el icono de la cámara.4. Pulsar en el botón aceptar.
Escenario alternativo	<ol style="list-style-type: none">1. Pulsar en la opción códigos QR del menú principal.2. Pulsar en la opción abrir seleccionar en el menú códigos QR.3. Seleccionar la foto del código QR guardada en la tarjeta de memoria.4. Pulsar en el botón aceptar.



Postcondiciones	<p>Pueden ocurrir dos cosas:</p> <ol style="list-style-type: none">Si la máquina ya está dada de alta en la base de datos: se muestra por pantalla los detalles de la misma.Si la máquina no está dada de alta en la base de datos: aparece un formulario con el nombre de la nueva máquina para poder añadirla.
------------------------	---

Tabla 20 - CU-008: Capturar código QR

CU-009: Consultar listado usuarios	
Descripción	Permite al usuario examinar un listado con todos los usuarios existentes en la base de datos de la aplicación.
Actores	Usuario de la aplicación.
Precondiciones	Iniciar la aplicación, estar logueado y que existan usuarios dados de alta en la base de datos.
Escenario normal	<ol style="list-style-type: none">Pulsar en la opción usuarios del menú principal.
Escenario alternativo	<p>No existe ningún usuario en la base de datos:</p> <ul style="list-style-type: none">Se muestra un mensaje indicándolo al usuario.
Postcondiciones	El listado es mostrado por pantalla al usuario.

Tabla 21 - CU-009: Consultar listado usuarios

CU-010: Consultar detalles de usuario	
Descripción	Permite al usuario examinar los detalles de un determinado usuario.
Actores	Usuario de la aplicación.
Precondiciones	Iniciar la aplicación, estar logueado y que existan usuarios dados de alta en la base de datos.
Escenario normal	<ol style="list-style-type: none">Pulsar en la opción usuarios del menú principal.Pulsar sobre el nombre de algún usuario del listado.
Escenario alternativo	-
Postcondiciones	Los datos de contacto del usuario son mostrados por pantalla al usuario.

Tabla 22 - CU-010: Consultar detalles de usuario



CU-011: Modificar datos usuario	
Descripción	Permite al usuario modificar los datos de un determinado usuario.
Actores	Usuario de la aplicación.
Precondiciones	Iniciar la aplicación, estar logueado y que existan usuarios dados de alta en la base de datos.
Escenario normal	<ol style="list-style-type: none">1. Pulsar en la opción usuarios del menú principal.2. Pulsar sobre el nombre de algún usuario del listado.3. Pulsar sobre el botón modificar en la parte inferior de los datos de contacto del usuario.4. Modificar aquellos campos que se considere oportunos.5. Pulsar sobre el botón modificar.
Escenario alternativo	<p>El formato de alguno de los campos modificados no es correcto:</p> <ul style="list-style-type: none">• Se muestra un mensaje indicándoselo al usuario.
Postcondiciones	Los datos del usuario son actualizados.

Tabla 23 - CU-011: Modificar datos usuario

CU-012: Eliminar usuario	
Descripción	Permite al usuario eliminar un usuario de la base de datos de la aplicación.
Actores	Usuario de la aplicación.
Precondiciones	Iniciar la aplicación, estar logueado y que existan usuarios dados de alta en la base de datos.
Escenario normal	<ol style="list-style-type: none">1. Pulsar en la opción usuarios del menú principal.2. Pulsar sobre el aspa que hay a la derecha de los nombres de usuario de algún usuario del listado.3. Pulsar sobre el botón aceptar.
Escenario alternativo	-
Postcondiciones	El usuario es eliminado de la base de datos.

Tabla 24 - CU-012: Eliminar usuario



CU-013: Añadir usuario	
Descripción	Permite al usuario añadir un usuario a la base de datos de la aplicación.
Actores	Usuario de la aplicación.
Precondiciones	Iniciar la aplicación y estar logueado.
Escenario normal	<ol style="list-style-type: none">1. Pulsar en la opción máquinas del menú principal.2. Pulsar en la opción añadir usuario del menú usuario.3. Completar los datos del formulario.4. Pulsar sobre el botón añadir.
Escenario alternativo	El formato de alguno de los campos no es correcto: <ul style="list-style-type: none">• Se muestra un mensaje indicándoselo al usuario.
Postcondiciones	El usuario es añadido a la base de datos.

Tabla 25 - CU-013: Añadir usuario

CU-014: Crear base de datos con máquinas de ejemplo	
Descripción	Permite al usuario crear una base de datos con máquinas de ejemplo, con el fin de comprobar el correcto funcionamiento de la aplicación sin tener que introducir las mismas manualmente.
Actores	Usuario de la aplicación.
Precondiciones	Iniciar la aplicación y estar logueado.
Escenario normal	<ol style="list-style-type: none">1. Pulsar en la opción administración en el menú principal.2. Seleccionar la opción Crear BD Máquinas del panel de administración.
Escenario alternativo	-
Postcondiciones	La base de datos se rellena con datos de máquinas ficticias.

Tabla 26 - CU-014: Crear base de datos con máquinas de ejemplo



CU-015: Crear base de datos con usuarios de ejemplo	
Descripción	Permite al usuario crear una base de datos con usuarios de ejemplo, con el fin de comprobar el correcto funcionamiento de la aplicación sin tener que introducir los mismos manualmente.
Actores	Usuario de la aplicación.
Precondiciones	Iniciar la aplicación y estar logueado.
Escenario normal	<ol style="list-style-type: none">1. Pulsar en la opción administración en el menú principal.2. Seleccionar la opción Crear BD Usuarios del panel de administración.
Escenario alternativo	-
Postcondiciones	La base de datos se rellena con datos de usuarios ficticios.

Tabla 27 - CU-015: Crear base de datos con usuarios de ejemplo

CU-016: Consultar log de la base de datos	
Descripción	Permite al usuario consultar un registro con todos los cambios que se han producido en la base de datos.
Actores	Usuario de la aplicación.
Precondiciones	Iniciar la aplicación, estar logueado y que exista algún cambio previo en la base de datos.
Escenario normal	<ol style="list-style-type: none">1. Pulsar en la opción log en el menú principal.
Escenario alternativo	No se ha hecho ninguna operación sobre la base de datos: <ul style="list-style-type: none">• Se muestra un mensaje indicándoselo al usuario.
Postcondiciones	El registro de los cambios producidos en la base de datos se muestra por pantalla al usuario.

Tabla 28 - CU-016: Consultar log de la base de datos



CU-017: Invertir orden eventos log de la base de datos	
Descripción	Permite seleccionar al usuario el orden en el que se mostrarán los diferentes eventos ocurridos en la base de datos: de más antiguo a más nuevo, o viceversa.
Actores	Usuario de la aplicación.
Precondiciones	Iniciar la aplicación, estar logueado y que existan 2 o más eventos en el log de la base de datos.
Escenario normal	<ol style="list-style-type: none">1. Pulsar en la opción log en el menú principal.2. Pulsar en el botón invertir log.
Escenario alternativo	-
Postcondiciones	El registro invierte su orden al pulsar el botón.

Tabla 29 - CU-017: Invertir orden eventos log de la base de datos

CU-018: Borrar base de datos de usuarios	
Descripción	Permite al usuario borrar los datos de los usuarios de la base de datos.
Actores	Usuario de la aplicación.
Precondiciones	Iniciar la aplicación y estar logueado. Que existan usuarios dados de alta en la base de datos.
Escenario normal	<ol style="list-style-type: none">1. Pulsar en la opción administración en el menú principal.2. Seleccionar la opción Borrar BD Usuarios del panel de administración.
Escenario alternativo	-
Postcondiciones	De la base de datos se eliminan todos los datos relacionados con los usuarios.

Tabla 30 - CU-018: Borrar base de datos de usuarios



CU-019: Borrar log de la base de datos	
Descripción	Permite al usuario borrar el log de la base de datos.
Actores	Usuario de la aplicación.
Precondiciones	Iniciar la aplicación y estar logueado. Que existan eventos en el log de la base de datos.
Escenario normal	<ol style="list-style-type: none">1. Pulsar en la opción administración en el menú principal.2. Seleccionar la opción Borrar Log del panel de administración.
Escenario alternativo	-
Postcondiciones	Se borran todos los eventos registrados en el log.

Tabla 31 - CU-019: Borrar log de la base de datos

CU-020: Borrar base de datos de máquinas	
Descripción	Permite al usuario borrar los datos de las máquinas de la base de datos.
Actores	Usuario de la aplicación.
Precondiciones	Iniciar la aplicación y estar logueado. Que existan máquinas dadas de alta en la base de datos.
Escenario normal	<ol style="list-style-type: none">1. Pulsar en la opción administración en el menú principal.2. Seleccionar la opción Borrar BD Máquinas del panel de administración.
Escenario alternativo	-
Postcondiciones	De la base de datos se eliminan todos los datos relacionados con las máquinas.

Tabla 32 - CU-020: Borrar base de datos de máquinas



3.2.2 Requisitos

En esta sección se expone el análisis del sistema realizado a través de los distintos tipos de requisitos. Estos han sido divididos en Requisitos Funcionales y Requisitos No Funcionales. En ambos casos, los atributos utilizados en los requisitos son los siguientes:

- **Identificador:** permite identificar inequívocamente a todos y cada uno de los requisitos a lo largo del proyecto. Se compone del tipo de requisito (RF para Requisitos Funcionales y RNF para Requisitos No Funcionales) y un número correlativo.
- **Nombre:** descripción corta que permite referirse al requisito de manera sencilla, y proporciona un resumen de su contenido.
- **Descripción:** contiene la condición que debe satisfacer el sistema para cumplir con el requisito.
- **Prioridad:** permite decidir qué requisitos deben implementarse antes o después. Los posibles valores son baja, media y alta.
- **Necesidad:** evalúa la importancia de este requisito, permitiendo establecer requisitos imprescindibles u opcionales. Los posibles valores son baja, media y alta.
- **Verificabilidad:** evalúa la dificultad de comprobar que el requisito está implementado. Los posibles valores son baja, media y alta.
- **Impacto:** evalúa el esfuerzo que supondría para el proyecto el cambio o eliminación del requisito. Los posibles valores son bajo, medio o alto.
- **Estabilidad:** evalúa la probabilidad de que el requisito cambie a lo largo del proyecto. Los posibles valores son baja, media y alta.

Los Requisitos Funcionales describen lo que el sistema debe hacer. Normalmente se identifican como los requisitos que responden a la pregunta ¿qué hace?

Los Requisitos No Funcionales son aquellos requerimientos que no se refieren directamente a las funciones detalladas que realiza el sistema, si no a aspectos de su diseño. Son los “recursos” necesarios para que trabaje el sistema de información. Normalmente se identifican como los requisitos que responden a la pregunta ¿cómo lo hace?

3.2.2.1 *Requisitos funcionales*

En esta sección se muestra la colección de requisitos funcionales que han sido extraídos durante la fase de análisis.



RF-001: Login			
Descripción	El usuario podrá loguearse en la aplicación.		
Prioridad	Alta	Necesidad	Alta
Verificabilidad	Alta	Impacto	Bajo
Estabilidad	Alta		

Tabla 33 - RF-001: Login

RF-002: Listar máquinas			
Descripción	El usuario podrá consultar un listado de las máquinas que hay dadas de alta en la base de datos.		
Prioridad	Alta	Necesidad	Alta
Verificabilidad	Alta	Impacto	Alto
Estabilidad	Alta		

Tabla 34 - RF-002: Listar máquinas

RF-003: Añadir máquina			
Descripción	El usuario podrá añadir una nueva máquina a la base de datos.		
Prioridad	Alta	Necesidad	Alta
Verificabilidad	Alta	Impacto	Alto
Estabilidad	Alta		

Tabla 35 - RF-003: Añadir máquina

RF-004: Gestión de solicitudes			
Descripción	El usuario podrá consultar las solicitudes hechas, así como aprobarlas o denegarlas.		
Prioridad	Media	Necesidad	Media
Verificabilidad	Alta	Impacto	Medio
Estabilidad	Alta		

Tabla 36 - RF-004: Gestión de solicitudes



RF-005: Información máquina			
Descripción	El usuario podrá ver en detalle la información de una determinada máquina.		
Prioridad	Alta	Necesidad	Alta
Verificabilidad	Alta	Impacto	Alto
Estabilidad	Alta		

Tabla 37 - RF-005: Información máquina

RF-006: Expandir campos información máquina			
Descripción	El usuario podrá expandir u ocultar cada uno de los campos de información de la máquina.		
Prioridad	Alta	Necesidad	Alta
Verificabilidad	Alta	Impacto	Alto
Estabilidad	Alta		

Tabla 38 - RF-006: Expandir campos información máquina

RF-007: Eliminar máquina			
Descripción	El usuario podrá eliminar máquina de la base de datos.		
Prioridad	Alta	Necesidad	Alta
Verificabilidad	Alta	Impacto	Alto
Estabilidad	Alta		

Tabla 39 - RF-007: Eliminar máquina

RF-008: Solicitar máquinas			
Descripción	El usuario podrá solicitar una máquina.		
Prioridad	Media	Necesidad	Media
Verificabilidad	Alta	Impacto	Alto
Estabilidad	Media		

Tabla 40 - RF-008: Solicitar máquinas



RF-009: Editar campos máquina			
Descripción	El usuario podrá editar cualquier campo asociado a una máquina, salvo el id.		
Prioridad	Media	Necesidad	Alta
Verificabilidad	Alta	Impacto	Medio
Estabilidad	Alta		

Tabla 41 - RF-009: Editar campos máquina

RF-010: Búsqueda de máquinas			
Descripción	El usuario podrá realizar búsquedas parciales e instantáneas en el listado de máquinas.		
Prioridad	Media	Necesidad	Alta
Verificabilidad	Alta	Impacto	Medio
Estabilidad	Alta		

Tabla 42 - RF-010: Búsqueda de usuarios

RF-011: Búsqueda de usuarios			
Descripción	El usuario podrá realizar búsquedas parciales e instantáneas en el listado de usuarios.		
Prioridad	Media	Necesidad	Alta
Verificabilidad	Alta	Impacto	Medio
Estabilidad	Alta		

Tabla 43 - RF-011: Búsqueda de usuarios

RF-012: Uso de códigos QR capturados por la cámara			
Descripción	El usuario podrá capturar códigos QR al instante a través de la cámara del dispositivo.		
Prioridad	Media	Necesidad	Media
Verificabilidad	Alta	Impacto	Medio
Estabilidad	Media		

Tabla 44 - RF-012: Uso de códigos QR capturados por la cámara



RF-013: Uso de códigos QR existentes en el álbum			
Descripción	El usuario podrá utilizar códigos QR ya existentes en el álbum del dispositivo.		
Prioridad	Media	Necesidad	Media
Verificabilidad	Alta	Impacto	Medio
Estabilidad	Media		

Tabla 45 - RF-013: Uso de códigos QR existentes en el álbum

RF-014: Listar usuarios			
Descripción	El usuario podrá consultar un listado de los usuarios que hay dados de alta en la base de datos.		
Prioridad	Alta	Necesidad	Alta
Verificabilidad	Alta	Impacto	Alto
Estabilidad	Alta		

Tabla 46 - RF-014: Listar usuarios

RF-015: Añadir usuario			
Descripción	El usuario podrá añadir un nuevo usuario a la base de datos.		
Prioridad	Alta	Necesidad	Alta
Verificabilidad	Alta	Impacto	Alto
Estabilidad	Alta		

Tabla 47 - RF-015: Añadir usuario

RF-016: Información usuario			
Descripción	El usuario podrá ver en detalle la información de un determinado usuario.		
Prioridad	Alta	Necesidad	Alta
Verificabilidad	Alta	Impacto	Alto
Estabilidad	Alta		

Tabla 48 - RF-016: Información usuario



RF-017: Eliminar usuario			
Descripción	El usuario podrá eliminar a un usuario de la base de datos.		
Prioridad	Alta	Necesidad	Alta
Verificabilidad	Alta	Impacto	Alto
Estabilidad	Alta		

Tabla 49 - RF-017: Eliminar usuario

RF-018: Editar campos usuario			
Descripción	El usuario podrá editar cualquier campo asociado a un usuario, salvo el id.		
Prioridad	Media	Necesidad	Alta
Verificabilidad	Alta	Impacto	Medio
Estabilidad	Alta		

Tabla 50 - RF-018: Editar campos usuario

RF-019: Histórico de los cambios en la base de datos			
Descripción	El usuario podrá consultar un histórico de los últimos cambios en la base de datos.		
Prioridad	Media	Necesidad	Alta
Verificabilidad	Alta	Impacto	Medio
Estabilidad	Alta		

Tabla 51 - RF-019: Histórico de los cambios en la base de datos

RF-020: Búsqueda en el histórico de cambios en la base de datos			
Descripción	El usuario podrá realizar búsquedas dentro del histórico de los cambios en la base de datos.		
Prioridad	Media	Necesidad	Alta
Verificabilidad	Alta	Impacto	Medio
Estabilidad	Alta		

Tabla 52 - RF-020: Búsqueda en el histórico de cambios en la base de datos



RF-021: Invertir orden del histórico de cambios en la base de datos			
Descripción	El usuario podrá invertir el orden en el que se muestra el histórico de los últimos cambios en la base de datos.		
Prioridad	Baja	Necesidad	Media
Verificabilidad	Alta	Impacto	Bajo
Estabilidad	Media		

Tabla 53 - RF-021: Invertir orden del histórico de cambios en la base de datos

RF-022: Borrar el histórico de cambios en la base de datos			
Descripción	El usuario podrá eliminar el histórico de los últimos cambios en la base de datos.		
Prioridad	Media	Necesidad	Alta
Verificabilidad	Alta	Impacto	Medio
Estabilidad	Alta		

Tabla 54 - RF-022: Borrar el histórico de cambios en la base de datos

RF-023: Crear base de datos de máquinas			
Descripción	Un usuario podrá crear una base de datos con datos de máquinas de ejemplo.		
Prioridad	Alta	Necesidad	Alta
Verificabilidad	Alta	Impacto	Medio
Estabilidad	Alta		

Tabla 55 - RF-023: Crear base de datos de máquinas

RF-024: Borrar base de datos de máquinas			
Descripción	Un usuario podrá borrar la tabla referente a los ordenadores de la base de datos.		
Prioridad	Alta	Necesidad	Alta
Verificabilidad	Alta	Impacto	Medio
Estabilidad	Alta		

Tabla 56 - RF-024: Borrar base de datos de máquinas



RF-025: Crear base de datos de usuarios			
Descripción	Un usuario podrá crear una base de datos con datos de usuarios de ejemplo.		
Prioridad	Alta	Necesidad	Alta
Verificabilidad	Alta	Impacto	Medio
Estabilidad	Alta		

Tabla 57 - RF-025: Crear base de datos de usuarios

RF-026: Borrar base de datos de usuarios			
Descripción	Un usuario podrá borrar la tabla referente a los usuarios de la base de datos.		
Prioridad	Alta	Necesidad	Alta
Verificabilidad	Alta	Impacto	Medio
Estabilidad	Alta		

Tabla 58 - RF-026: Borrar base de datos de usuarios

RF-027: Logout			
Descripción	El usuario podrá desloguearse de la aplicación.		
Prioridad	Alta	Necesidad	Alta
Verificabilidad	Alta	Impacto	Bajo
Estabilidad	Alta		

Tabla 59 - RF-027: Logout

3.2.2.2 *Requisitos no funcionales*

En esta sección se muestra la colección de requisitos no funcionales que han sido extraídos durante la fase de análisis.



RNF-001: La aplicación se desarrollará en Phonegap			
Descripción	La aplicación se desarrollará en Phonegap para permitir el uso en dispositivos con distintos SO.		
Prioridad	Alta	Necesidad	Alta
Verificabilidad	Alta	Impacto	Alto
Estabilidad	Alta		

Tabla 60 - RNF-001: La aplicación se desarrollará en Phonegap

RNF-002: La interfaz de la aplicación se gestionará mediante jQuery Mobile			
Descripción	La interfaz se creará con el <i>framework</i> jQuery Mobile.		
Prioridad	Alta	Necesidad	Alta
Verificabilidad	Media	Impacto	Alto
Estabilidad	Alta		

Tabla 61 - RNF-002: La interfaz de la aplicación se gestionará mediante jQuery Mobile

RNF-003: El soporte de almacenamiento será WebSQL			
Descripción	El almacenamiento de los datos de la aplicación se gestionará mediante WebSQL.		
Prioridad	Alta	Necesidad	Alta
Verificabilidad	Baja	Impacto	Medio
Estabilidad	Media		

Tabla 62 - RNF-003: El soporte de almacenamiento será WebSQL

RNF-004: El decodificador QR se gestionará de manera interna			
Descripción	Se deberá decodificar los códigos QR desde la propia aplicación sin necesidad de utilizar aplicaciones de terceros.		
Prioridad	Alta	Necesidad	Media
Verificabilidad	Media	Impacto	Medio
Estabilidad	Alta		

Tabla 63 - RNF-004: El decodificador QR se gestionará de manera interna



3.2.1 Matriz de trazabilidad Casos de Uso – Requisitos

En esta sección se mostrará la matriz de trazabilidad entre los casos de uso y los requisitos vistos anteriormente.

	RF-001	RF-002	RF-003	RF-004	RF-005	RF-006	RF-007	RF-008	RF-009	RF-010	RF-011	RF-012	RF-013	RF-014	RF-015	RF-016	RF-017	RF-018	RF-019	RF-020	RF-021	RF-022	RF-023	RF-024	RF-025	RF-026
CU-01		X								X																
CU-02		X			X	X																				
CU-03		X			X			X																		
CU-04		X			X				X																	
CU-05		X					X																			
CU-06			X																							
CU-07				X																						
CU-08			X		X							X	X													
CU-09											X			X												
CU-10														X		X										
CU-11														X		X		X								
CU-12														X			X									
CU-13															X											
CU-14																							X			
CU-15																									X	
CU-16																				X						
CU-17																		X			X					
CU-18																X										
CU-19																						X				
CU-20																								X		

Tabla 64 - Matriz de trazabilidad Casos de Uso - Requisitos



3.3 Diseño

En este apartado se mostrará el diseño del sistema. Se detallará el diagrama de navegación de ventanas de la aplicación desarrollada. Se podrá comprobar el esquema de la base de datos que soporta la aplicación, así como una breve descripción del mismo. Por último, se mostrará el diseño de las pruebas que ha de superar la aplicación para satisfacer los requerimientos vistos en la sección anterior.

3.3.1 Diagramas de navegación

En esta sección se muestra el diagrama de navegación de ventanas de la aplicación. Aparte de lo indicado en el diagrama, desde cualquier ventana se puede volver directamente a *Main Page* pulsando el botón *HOME*. Además, desde *Computer Details* y *User Details* se puede volver a la pantalla precedente pulsando el botón Atrás. Por último, se puede hacer *logout* pulsando la tecla atrás desde *Main Page*.

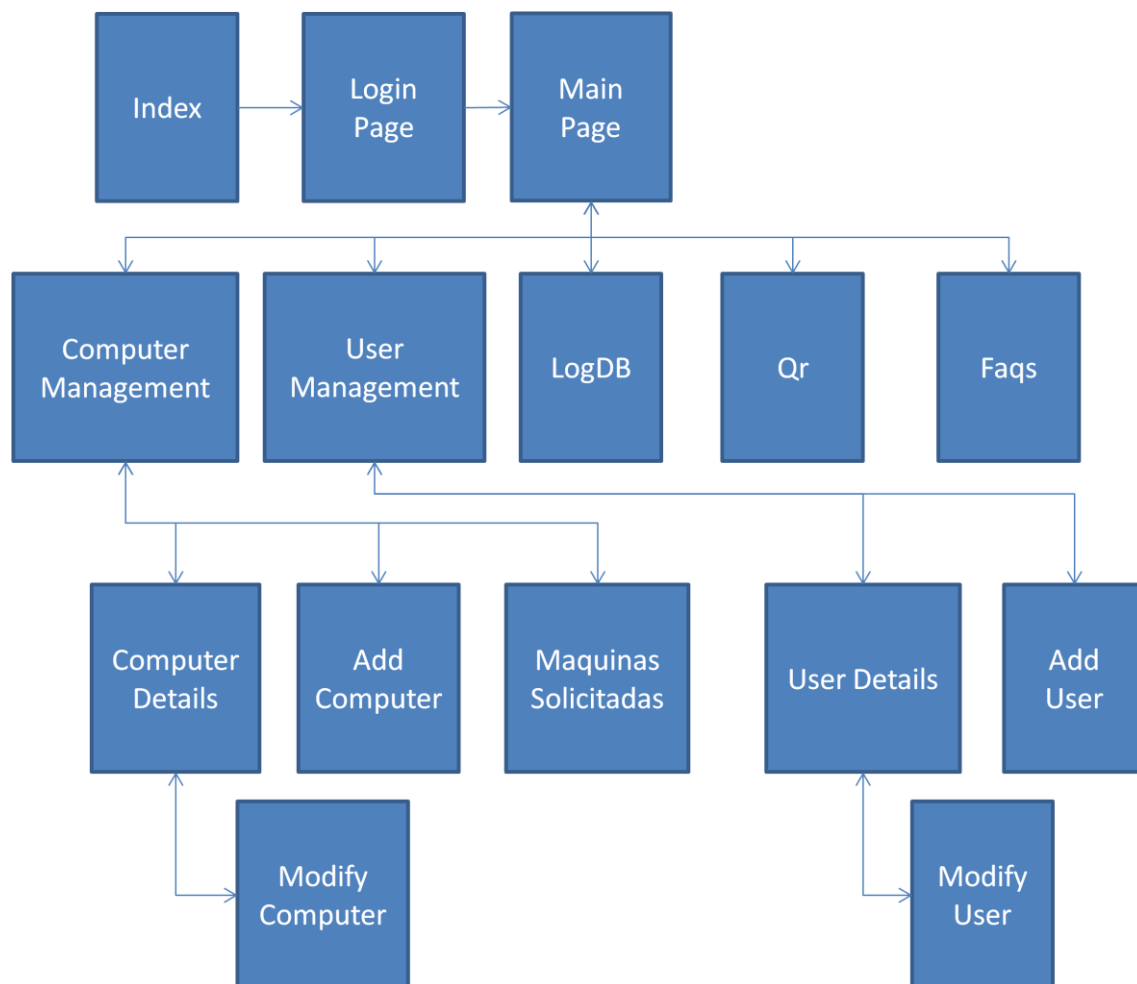


Ilustración 15 - Diagrama de navegación de ventanas



3.3.2 Esquema de la base de datos

En esta sección se incluirá el esquema de la base de datos, así como una breve descripción del mismo.

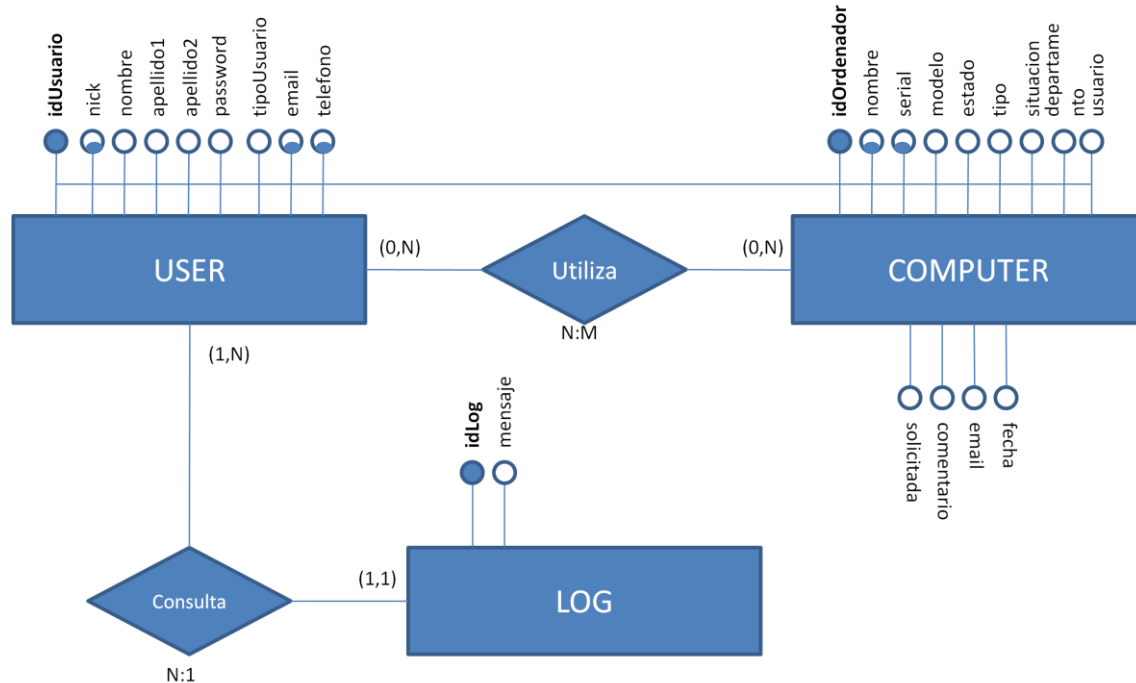


Ilustración 16 - Modelo ER de la base de datos

3.3.2.1 Tabla USERS

A continuación se explicarán cada uno de los atributos de la tabla USERS:

- idUsuario: identificador único para cada usuario. Es de tipo INT, clave primaria y autoincremental.
- nick: nick del usuario. Es de tipo TEXT, único y con una longitud máxima de 10 caracteres.
- nombre: nombre de pila del usuario. Es de tipo TEXT con una longitud máxima de 10 caracteres.
- apellido1: primer apellido del usuario. Es de tipo TEXT con una longitud máxima de 15 caracteres.
- apellido2: segundo apellido del usuario. Es de tipo TEXT con una longitud máxima de 15 caracteres.
- passwordD contraseña del usuario. Es de tipo TEXT con una longitud mínima de 6 caracteres y una máxima de 12. Se almacena cifrada mediante SHA1.
- tipoUsuario: tipo de usuario. Es de tipo TEXT, con una longitud máxima de 15 caracteres. En el prototipo actual no afecta al funcionamiento de la aplicación, pero



en el futuro habrá distintos tipos de usuario, con roles que les permitirán realizar diferentes acciones.

- email: correo electrónico del usuario. Es de tipo TEXT, único y con una longitud máxima de 30 caracteres.
- telefono: número de teléfono del usuario. Es de tipo INT, único y con una longitud máxima de 9 dígitos.

3.3.2.2 Tabla COMPUTERS

A continuación se explicarán cada uno de los atributos de la tabla COMPUTERS:

- idOrdenador: identificador único para cada ordenador. Es de tipo INT, clave primaria y autoincremental.
- nombre: nombre asignado al ordenador. Es de tipo TEXT, único y con una longitud máxima de 10 caracteres.
- serial: número de serie del ordenador. Es de tipo TEXT, único y con una longitud máxima de 15 caracteres.
- modelo: modelo del ordenador. Es de tipo TEXT, con una longitud máxima de 10 caracteres.
- estado: estado en el que se encuentra el ordenador. Es de tipo TEXT con una longitud máxima de 20 caracteres.
- tipo: tipo de ordenador. Tipo TEXT con una longitud máxima de 20 caracteres.
- situacion: lugar en el que se encuentra el ordenador. Es de tipo TEXT con una longitud máxima de 15 caracteres.
- departamento: departamento al que se le asigna el ordenador. Es de tipo TEXT con una longitud máxima de 15 caracteres.
- usuario: persona a la que se le asigna el ordenador. Es de tipo INT y es clave ajena de la tabla USERS.
- fecha: fecha en la que se da de alta el ordenador. Es de tipo DATE.
- email: correo electrónico de contacto. Es de tipo TEXT con una longitud máxima de 30 caracteres.
- comentario: comentarios sobre la máquina. Es de tipo TEXT y está limitado a 100 caracteres por comentario, mostrando un historial de todos los comentarios.
- solicitada: *flag* para indicar si una máquina ha sido solicitada o no. Es de tipo INT y está limitado a valer 0/1.

3.3.2.3 Tabla LOG

A continuación se explicarán cada uno de los atributos de la tabla LOG:

- idLog: identificador único para cada entrada del *log*. Es de tipo INT, clave primaria y autoincremental.
- mensaje: mensaje que se mostrará en el *log* con los cambios sufridos por la base de datos. Es de tipo TEXT con una longitud máxima de 100 caracteres.



3.3.3 Diseño de las pruebas

En esta sección se detallarán las pruebas que se realizarán para el sistema desarrollado. Se ha decidido que el único tipo de pruebas que se van a realizar son pruebas de aceptación. Se trata de un conjunto de pruebas funcionales realizadas por el usuario final sobre el sistema implantado. En este caso el usuario final que deberá validar las pruebas es el cliente. Se nombra a un responsable de pruebas que ha de certificar que las mismas se han pasado correctamente. Tienen como finalidad validar el comportamiento del sistema y verificar sus funcionalidades. Son las más importantes, puesto que demuestran que la funcionalidad está terminada y el sistema funciona correctamente. Los atributos utilizados en la definición de las pruebas son los siguientes:

- **Identificador:** permite identificar inequívocamente a todas y cada una de las pruebas a lo largo del proyecto. Se compone del tipo de prueba (PA para Pruebas de Aceptación) y un número correlativo.
- **Descripción:** contiene la condición que debe satisfacer el sistema para pasar la prueba.
- **Requisitos relacionados:** identifica aquellos requisitos que están relacionados con la prueba que se plantea.

PA-001	
Descripción	Comprobar que el usuario puede loguearse en la aplicación correctamente.
Requisitos relacionados	RF-001

Tabla 65 - PA-001

PA-002	
Descripción	Comprobar que el usuario puede consultar un listado de las máquinas existentes en la base de datos.
Requisitos relacionados	RF-002

Tabla 66 - PA-002

PA-003	
Descripción	Comprobar que el usuario puede añadir una nueva máquina a la base de datos.
Requisitos relacionados	RF-003

Tabla 67 - PA-003



PA-004	
Descripción	Comprobar que el usuario puede consultar un listado con aquellas máquinas que han sido solicitadas.
Requisitos relacionados	RF-004

Tabla 68 - PA-004

PA-005	
Descripción	Comprobar que el usuario puede aprobar o rechazar la solicitud de una determinada máquina.
Requisitos relacionados	RF-004

Tabla 69 - PA-005

PA-006	
Descripción	Comprobar que el usuario puede consultar los detalles de una determinada máquina.
Requisitos relacionados	RF-005

Tabla 70 - PA-006

PA-007	
Descripción	Comprobar que el usuario puede expandir cada uno de los campos de una determinada máquina.
Requisitos relacionados	RF-006

Tabla 71 - PA-007

PA-008	
Descripción	Comprobar que el usuario puede eliminar una determinada máquina de la base de datos.
Requisitos relacionados	RF-007

Tabla 72 - PA-008



PA-009	
Descripción	Comprobar que el usuario puede solicitar una determinada máquina.
Requisitos relacionados	RF-008

Tabla 73 - PA-009

PA-010	
Descripción	Comprobar que el usuario puede modificar los datos de una determinada máquina.
Requisitos relacionados	RF-009

Tabla 74 - PA-010

PA-011	
Descripción	Comprobar que el usuario puede capturar códigos QR mediante la cámara.
Requisitos relacionados	RF-012

Tabla 75 - PA-011

PA-012	
Descripción	Comprobar que el usuario puede abrir códigos QR existentes en la galería de imágenes.
Requisitos relacionados	RF-013

Tabla 76 - PA-012

PA-013	
Descripción	Comprobar que el usuario puede consultar un listado de los usuarios existentes en la base de datos.
Requisitos relacionados	RF-014

Tabla 77 - PA-013



PA-014	
Descripción	Comprobar que el usuario puede añadir un nuevo usuario a la base de datos.
Requisitos relacionados	RF-015

Tabla 78 - PA-014

PA-015	
Descripción	Comprobar que el usuario puede consultar los detalles de un determinado usuario en la base de datos.
Requisitos relacionados	RF-016

Tabla 79 - PA-015

PA-016	
Descripción	Comprobar que el usuario puede eliminar un determinado usuario de la base de datos.
Requisitos relacionados	RF-017

Tabla 80 - PA-016

PA-017	
Descripción	Comprobar que el usuario puede modificar los datos de un determinado usuario.
Requisitos relacionados	RF-018

Tabla 81 - PA-017

PA-018	
Descripción	Comprobar que el usuario puede consultar un log de los cambios que han tenido lugar en la base de datos.
Requisitos relacionados	RF-019

Tabla 82 - PA-018



PA-019	
Descripción	Comprobar que el usuario pueda invertir el orden en el que es mostrado el log.
Requisitos relacionados	RF-021

Tabla 83 - PA-019

PA-020	
Descripción	Comprobar que el usuario puede borrar el log.
Requisitos relacionados	RF-022

Tabla 84 - PA-020

PA-021	
Descripción	Comprobar que el usuario puede crear una base de datos con máquinas de ejemplo.
Requisitos relacionados	RF-023

Tabla 85 - PA-021

PA-022	
Descripción	Comprobar que el usuario puede crear una base de datos con usuarios de ejemplo.
Requisitos relacionados	RF-025

Tabla 86 - PA-022

PA-023	
Descripción	Comprobar que el usuario puede borrar la base de datos de máquinas.
Requisitos relacionados	RF-024

Tabla 87 - PA-023



PA-024	
Descripción	Comprobar que el usuario puede borrar la base de datos de usuarios.
Requisitos relacionados	RF-026

Tabla 88 - PA-024

PA-025	
Descripción	Comprobar que un usuario puede realizar búsquedas parciales e instantáneas en los diferentes listados de la aplicación.
Requisitos relacionados	RF-002, RF-004, RF-010, RF-011, RF-014, RF-020

Tabla 89 - PA-025

PA-026	
Descripción	Comprobar que no permite loguearse a un usuario inexistente o cuya contraseña no sea correcta.
Requisitos relacionados	RF-001

Tabla 90 - PA-026

PA-027	
Descripción	Comprobar que no permite introducir datos erróneos en los formularios.
Requisitos relacionados	RF-003, RF-009, RF-015, RF-018

Tabla 91 - PA-027

PA-028	
Descripción	Comprobar que el usuario puede desloguearse de la aplicación.
Requisitos relacionados	RF-027

Tabla 92 - PA-028

En el apartado Informes de los casos de prueba se comprobarán los resultados de las pruebas definidas en este apartado.



3.3.3.1 Matriz de trazabilidad Pruebas – Requisitos

En esta sección se mostrará la matriz de trazabilidad entre las pruebas de aceptación y los requisitos vistos anteriormente.

	PA-01	PA-02	PA-03	PA-04	PA-05	PA-06	PA-07	PA-08	PA-09	PA-10	PA-11	PA-12	PA-13	PA-14	PA-15	PA-16	PA-17	PA-18	PA-19	PA-20	PA-21	PA-22	PA-23	PA-24	PA-25	PA-26	PA-27	PA-28
RF-001	X																									X		
RF-002		X																							X			
RF-003			X																								X	
RF-004				X	X																				X			
RF-005						X																						
RF-006							X																					
RF-007								X																				
RF-008									X																			
RF-009										X																		X
RF-010																									X			
RF-011																									X			
RF-012											X																	
RF-013												X																
RF-014													X												X			
RF-015														X													X	
RF-016															X													
RF-017																X												
RF-018																	X										X	
RF-019																		X										
RF-020																									X			
RF-021																			X									
RF-022																				X								
RF-023																					X							
RF-024																							X					
RF-025																						X						
RF-026																								X				
RF-027																												X

Ilustración 17 - Matriz de trazabilidad Pruebas de Aceptación y Requisitos



3.4 Implementación

En este apartado se comentarán los aspectos que más ha costado realizar durante la implementación de la aplicación.

Para descargar la aplicación, se puede seguir el siguiente enlace:

<https://www.dropbox.com/s/i95zxo8ee8r8pri/iNventariator.apk>

Para acceder al repositorio con el código fuente de la aplicación, se puede seguir el siguiente enlace:

https://www.dropbox.com/sh/ev9jjv5ctimb2oh/AACU2TixeEerW_A6MD7rw0JPa

Si bien no ha habido ningún punto bloqueante durante su desarrollo, hay ciertos aspectos que, por unas causas u otras, ha costado más llevar a cabo. Cabe destacar los siguientes:

- Refresco de las listas: al generar el contenido de las listas del HTML en el Javascript, los datos aparecían correctamente, pero no seguían el formato jQuery Mobile, es decir, no adoptaba el estilo de las CSS definidas en este estándar. Esto ocurrió al principio de la fase de implementación, en la que aún estaba familiarizándome con jQuery Mobile. Tras investigar, descubrí que el error se basaba en que era necesario refrescar la lista para que cogiera de nuevo el formato correcto. Para ello se utiliza la siguiente línea de código:

```
$("#ulId").listview('refresh');
```

- Refresco del resto de elementos: una vez descubierto (y solucionado) el problema anterior, me di cuenta de que cada elemento tiene sus propios métodos para refrescarse. Algunos, como los *collapsibles*, no supusieron un gran problema. Otros como preseleccionar una opción en un *input* de tipo radio en un formulario dieron más problemas, ya que asignaba bien el valor pero no conseguía que dicha opción apareciera ya marcada (por ejemplo en los formularios de modificar máquina).

```
//Refresco de botones
$(".classButton").button();
//Refresco de collapsibles
$(".classCollapsible").collapsible();
//Refresco de valor input seleccionable formulario
$("#classInput").selectmenu("refresh");
//Refresco de valor input radio button
$("#classInput").checkboxradio("refresh");
```

- Decodificación de los códigos QR: como se ha comentado en el apartado de Elementos utilizados, hubo una serie de problemas con la decodificación de los códigos QR. Finalmente, se consiguió hacer funcionar el *Plugin Lazar Soft*. Aún así



también hubo que superar ciertos problemas hasta que se consiguió que funcionara. El primer problema que hubo es que no encontraba ciertas funciones al hacer las llamadas, a pesar de haber incluido la librería. Este problema estaba originado porque en el HTML desde el que se llama al código que gestiona los códigos QR había que incluir además 17 librerías adicionales y en un orden concreto. Tras solventar el mismo, hubo otro problema, y es que tras hacer la lectura del código QR no se llamaba correctamente a la función *callback* que gestionaba la imagen. Tras localizar el error, se modificó el código y comenzó a funcionar correctamente. Finalmente, el último problema que hubo que solventar, es que al producirse un error en la decodificación de un código QR, se notificaba al usuario, pero aún así se redirigía al formulario de añadir máquina y en el nombre aparecía el código de error. Esto se solucionó modificando la redirección de la función error.

- Cifrado de datos: como se ha comentado en el apartado de Elementos utilizados, se ha decidido cifrar algunos de los datos almacenados. Debido a la naturaleza de la aplicación, se ha decidido cifrar únicamente las contraseñas de acceso al sistema. Para ello, la contraseña no es almacenada como tal, sino que se almacena su *hash* SHA1. En la pantalla de *login*, se compara el *hash* de la contraseña introducida por el usuario con el *hash* almacenado en la base de datos. De este modo, si un atacante interceptara la comunicación del formulario con la base de datos, o si consiguiera acceder a la base de datos, obtendría el valor del *hash* de la contraseña, lo cual no le serviría de nada, puesto que si introdujera dicho valor en el campo contraseña de la pantalla de *login*, se calcularía el *hash* del *hash*, y no coincidiría con el valor de la base de datos, tal y como se puede observar en la Ilustración 18. El motivo por el que no se han cifrado mediante este método el resto de campos de la base de datos, es que se trata de una función irreversible, es decir, no se puede descifrar el texto una vez cifrado, por lo que no se podrían mostrar los datos al resultar ilegibles. Tampoco se han cifrado los datos utilizando algoritmos que permitan descifrar porque al tener que descifrar los datos cada vez que quieran ser mostrados, la aplicación perdería fluidez, empeorando la eficiencia de uso, y por tanto, uno de los pilares básicos de la aplicación.

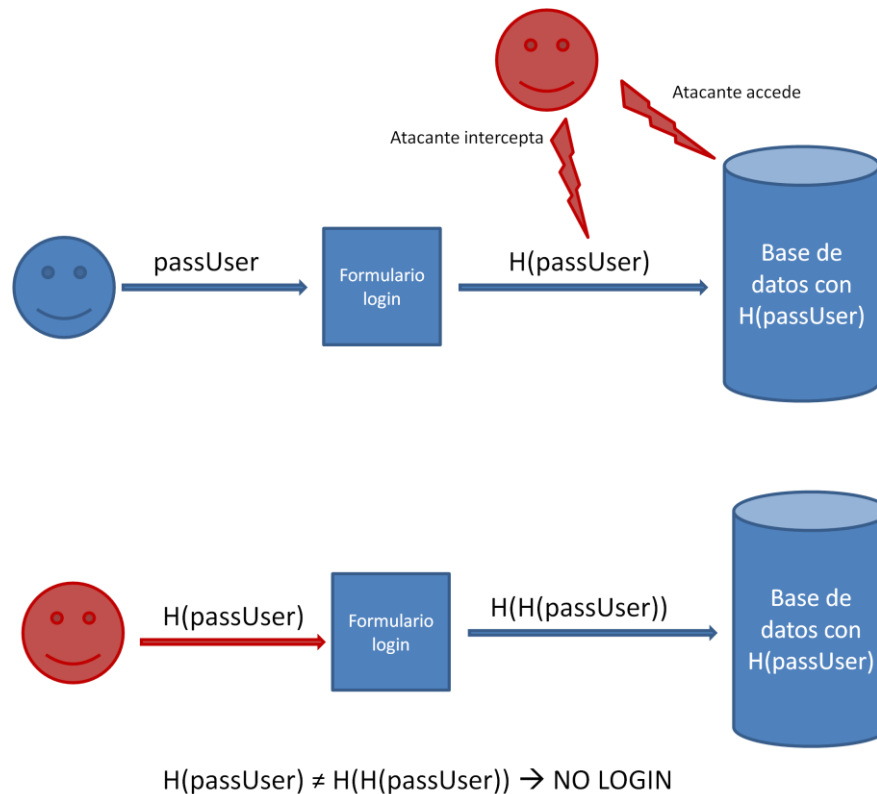


Ilustración 18 - Esquema de cifrado de contraseña

Para utilizar este cifrado, simplemente se incluye la librería `sha1.js` en el proyecto, y se llama a la siguiente función:

```
hex_sha1("Texto a cifrar").
```

Para verificar su correcto funcionamiento, se ha probado a obtener la codificación `hash` vacía, ya que se sabe que corresponde a la siguiente:

```
SHA1("") = da39a3ee5e6b4b0d3255bfe95601890afd80709
```

3.5 Implantación

En este apartado se resumirá la puesta en marcha del sistema así como las explicaciones sobre el manejo del mismo.

3.5.1 Instalación en sistema Android

Para instalar la aplicación basta con descargar el `.apk` en la tarjeta de memoria del *Smartphone*. En caso de querer comprobar la veracidad del `.apk` descargado, se puede calcular su suma de verificación (*checksum*). Esta suma es única, por lo que con ello se asegura que el contenido del `.apk` no ha sido modificado por terceros, con el fin de incluir código malicioso u otras intenciones, o que el contenido de la descarga es incompleto o



Análisis, diseño e implementación de un sistema para ayudar en el control de inventario.
Capítulo III: Análisis, diseño, implementación e implantación.

está corrupto. Para comprobar la suma de verificación del archivo, se han de seguir los siguientes pasos:

- Descargar el .apk en el ordenador.
- En caso de ser un entorno Windows:
 - Seguir los pasos del apartado *Installation* en el soporte de Microsoft [23].
 - Una vez instalada la utilidad, se puede calcular el *checksum* con la siguiente línea:

```
FCIV -md5 -sha1 path\filename.ext
```

*Nota: en caso de copiar el archivo en C:\FCIV, sólo hace falta poner *filename.ext* en lugar de la ruta completa.

La ejecución del comando nos devolverá la huella MD5 y la huella SHA1.

- En caso de ser un entorno Mac:
 - Abrir un terminal y ejecutar las siguientes líneas [24]:

```
openssl md5 [full path to file]  
openssl sha1 [full path to file]
```

- En caso de ser un entorno Linux:
 - Abrir un terminal y ejecutar las siguientes líneas [25]:

```
md5sum [full path to file]  
sha1sum [full path to file]
```

Las huellas calculadas han de corresponderse con las siguientes:

```
//Huella MD5  
c14149ac1fa0a508e1b001a522dc2498  
  
//Huella SHA-1  
fc9f87ec82e640bacc4bd7fab406b55a882221a5
```

En caso de coincidencia, se puede asegurar que el archivo no ha sido alterado y se corresponde con el original.

Una vez tenemos el .apk en la tarjeta de memoria, se ha de seleccionar y dar al botón de instalar. Aparecerán los permisos que son necesarios aceptar para su instalación. Tras aceptarlos, se completa la instalación.

3.5.2 Manual de usuario

Para consultar el manual de uso de la aplicación, se puede consultar el Anexo I del presente documento.



4 Planificación del proyecto

En este capítulo se detallará la planificación inicial y el desarrollo real del Trabajo de Fin de Grado, así como el estudio de las desviaciones producidas entre ambos. Para ello, se incluirá un diagrama de Gantt en cada uno de los estudios. A partir de esta planificación, se detallará un presupuesto en el capítulo 6, teniendo en cuenta el coste de personal, coste de material, costes indirectos, etc.

4.1 Planificación inicial

En este apartado se puede ver la planificación que se ha realizado inicialmente para el desarrollo del Trabajo de Fin de Grado. En esta planificación se han incluido todas las fases en las que se divide el proyecto, junto con una estimación del esfuerzo a realizar para cada una de ellas.

Para la elaboración de la planificación del Trabajo de Fin de Grado se ha tenido en cuenta que no se puede dedicar 8 horas al día, ya que la realización del mismo se ha tenido que compaginar con una jornada laboral de 5 horas diarias.

El plazo inicial que se establece para la realización del Trabajo de Fin de Grado es del lunes 27 de enero de 2014 (fecha de inicio del 2º cuatrimestre) hasta el viernes 6 de junio de 2014 (15 días antes del cierre del plazo para la entrega del Trabajo de Fin de Grado), con una dedicación de 4 horas diarias, lo que deja un total de 88 días (fines de semana y festivos no incluidos), y un total de 352 horas.

A continuación se puede observar una tabla resumen, con la estimación de esfuerzo para cada una de las fases.

Fase	Días	Horas	% Total
Trabajo de Fin de Grado	88 días	352 horas	100,00%
Planificación Inicial	03 días	12 horas	3,41%
Estudio estado de la cuestión	04 días	16 horas	4,55%
Estudio de soluciones similares	02 días	08 horas	2,27%
Estudio de elementos a utilizar	02 días	08 horas	2,27%
Análisis	17 días	68 horas	19,32%
Especificación de casos de uso	06 días	24 horas	6,82%
Especificación de requisitos	11 días	44 horas	12,50%



Fase	Días	Horas	% Total
Diseño	08 días	32 horas	9,09%
Diagramas de navegación	02 días	08 horas	2,27%
Diagramas de la base de datos	02 días	08 horas	2,27%
Diseño de las pruebas	04 días	16 horas	4,55%
Implementación	43 días	172 horas	48,86%
Pruebas	04 días	16 horas	4,55%
Documentación	09 días	36 horas	10,23%

Tabla 93 - Planificación inicial del proyecto

Como se puede observar en la tabla anterior, la mayor parte del tiempo (prácticamente la mitad) se invertirá en la implementación de la aplicación. De mayor a menor, las fases de diseño y análisis son las siguientes que más tiempo llevarán, seguida de la documentación. El resto de fases llevarán menos tiempo.

Aunque la documentación se haga al final del proyecto, todo lo necesario para confeccionarla (estudio del estado de la cuestión, análisis, diseño y pruebas) se ha realizado previamente, es decir, en la fase de documentación simplemente se documenta cada una de las fases (por ejemplo, los casos de uso o el catálogo de requisitos existen desde la fase en la que se han realizado, bien en borradores o documentos aparte).

Como se ha comentado en el apartado de Metodología utilizada, las fases de Análisis, Diseño, Implementación y Pruebas no se han desarrollado en cascada, si no que se ha aplicado el concepto de metodología ágil para hacerlo de un modo iterativo e incremental. Tanto los datos de la Tabla 93 como la representación en el diagrama de Gantt de la Ilustración 19 para dichas fases se tratan de la suma de todos los tiempos parciales que ha llevado realizar el Análisis, Diseño, Implementación y Pruebas para los casos de uso y requisitos que se han realizado en cada uno de los *sprints*. El motivo de que la planificación se represente de esta manera, es poder reutilizar el conocimiento adquirido con la realización del presente proyecto en futuros proyectos, sin necesidad de aplicar metodologías ágiles, cuando la experiencia permita poder realizarlo de este modo.

A continuación se puede observar un diagrama de Gantt con la información relativa a la planificación del proyecto.



Análisis, diseño e implementación de un sistema para ayudar en el control de inventario. Capítulo IV: Planificación del proyecto.

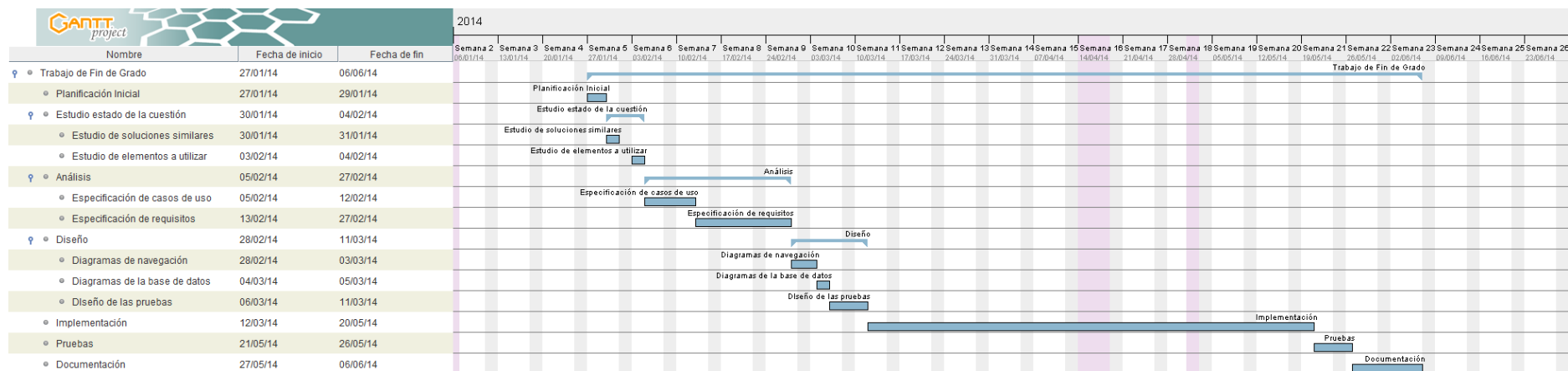


Ilustración 19 - Diagrama de Gantt de la planificación inicial del proyecto



4.2 Desarrollo real

En este apartado se puede ver el desarrollo real que ha tenido el Trabajo de Fin de Grado. En este desarrollo se han incluido todas las fases en las que finalmente se ha dividido el proyecto, junto con el esfuerzo que ha llevado realizar cada una de ellas.

Como se ha comentado en la sección de Planificación inicial, el desarrollo del proyecto se ha tenido que compaginar con una jornada laboral de 5 horas diarias.

El plazo que finalmente ha habido para la realización del Trabajo de Fin de Grado es del lunes 20 de enero de 2014 (1 semana antes de la fecha planificada inicialmente) hasta el miércoles 18 de junio de 2014 (4 días antes del cierre del plazo para la entrega del Trabajo de Fin de Grado), con una dedicación de 4 horas diarias, lo que deja un total de 101 días (fines de semana y festivos no incluidos), y un total de 404 horas.

A continuación se puede observar una tabla resumen, con el esfuerzo invertido en cada una de las fases.

Fase	Días	Horas	% Total
Trabajo de Fin de Grado	101 días	404 horas	100,00%
Planificación Inicial	02 días	08 horas	1,98%
Estudio estado de la cuestión	04 días	16 horas	3,96%
Estudio de soluciones similares	01 días	04 horas	0,99%
Estudio de elementos a utilizar	03 días	12 horas	2,97%
Análisis	19 días	76 horas	18,81%
Especificación de casos de uso	09 días	36 horas	8,91%
Especificación de requisitos	10 días	40 horas	9,90%
Diseño	08 días	32 horas	7,92%
Diagramas de navegación	03 días	12 horas	2,97%
Diagramas de la base de datos	02 días	08 horas	1,98%
Diseño de las pruebas	03 días	12 horas	2,97%
Preparación del sistema	05 días	20 horas	4,95%
Instalación de Eclipse	01 días	02 horas	0,50%



Fase	Días	Horas	% Total
Configuración de Android Developer Tools	01 días	02 horas	0,50%
Configuración de jQuery + jQuery Mobile	01 días	02 horas	0,50%
Configuración de Phonegap	01 días	02 horas	0,50%
Aprendizaje de jQuery y jQuery Mobile	01 días	04 horas	0,99%
Aprendizaje de Phonegap	02 días	06 horas	1,49%
Compilación de HolaMundo en Phonegap	01 días	02 horas	0,50%
Implementación	51 días	204 horas	50,50%
Pruebas	02 días	08 horas	1,98%
Documentación	10 días	40 horas	9,90%

Tabla 94 - Desarrollo real del proyecto

Como se puede observar en la tabla anterior, el tiempo real que ha llevado desarrollar el proyecto es superior al inicialmente estimado, si bien las tareas mantienen aproximadamente el porcentaje de tiempo sobre el total. Además, con el fin de intentar finalizar el proyecto en la fecha planificada, se ha tenido que adelantar el inicio de la realización del Trabajo de Fin de Grado, ya que durante la planificación inicial no se habían tenido en cuenta las tareas de preparación del sistema.

Al igual que lo que se ha comentado en el apartado de Planificación inicial, tanto los datos de la Tabla 94Tabla 93 como la representación en el diagrama de Gantt de la Ilustración 20 para dichas fases se tratan de la suma de todos los tiempos parciales que ha llevado realizar el Análisis, Diseño, Implementación y Pruebas para los casos de uso y requisitos que se han realizado en cada uno de los *sprints*. La fase de Preparación del sistema se ha realizado al finalizar el análisis y el diseño del primer *sprint*, aunque en el diagrama se ha colocado justo antes de la fase de Implementación, como si hubiera sido un desarrollo en cascada.

A continuación se puede observar un diagrama de Gantt con la información relativa al desarrollo real del proyecto.



Análisis, diseño e implementación de un sistema para ayudar en el control de inventario.
 Capítulo IV: Planificación del proyecto.

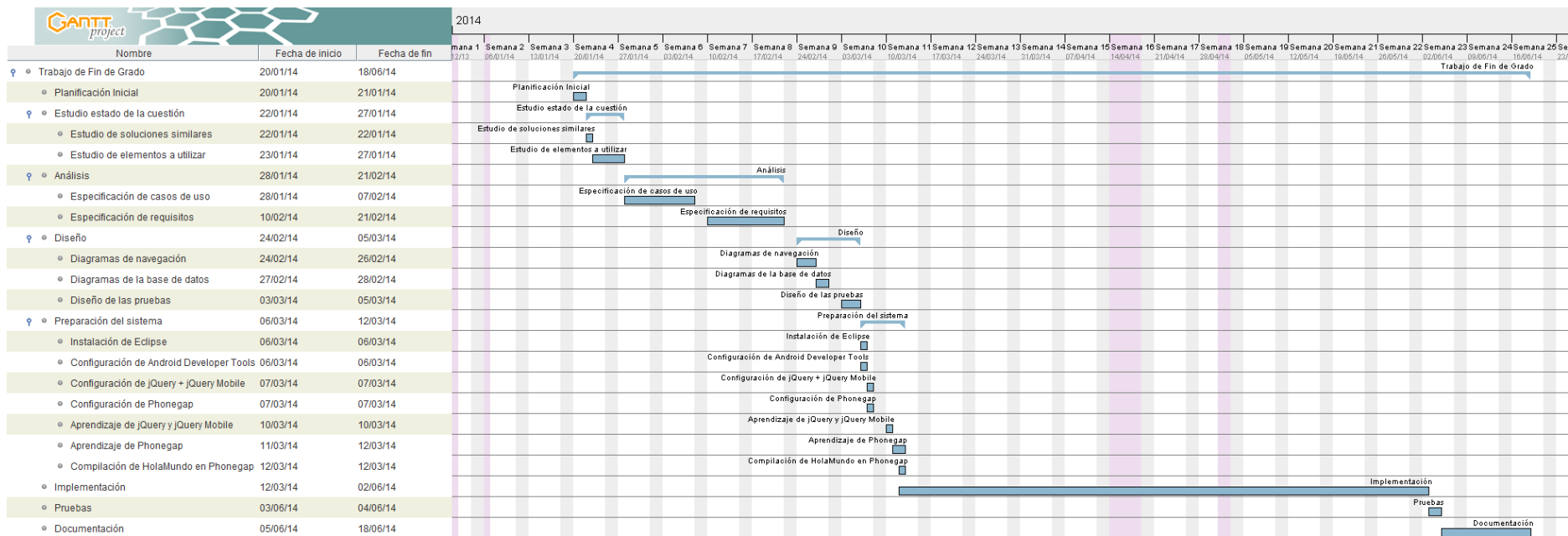


Ilustración 20 - Diagrama de Gantt del desarrollo real del proyecto



4.3 Comparativa entre la planificación inicial y el desarrollo real

En este apartado se compara la planificación inicial con el desarrollo real que finalmente ha tenido el Trabajo de Fin de Grado, con el fin de estudiar las desviaciones producidas durante la realización del mismo. A continuación se puede observar una tabla resumen, con la comparativa anteriormente citada.

Fase	Días P.Inicial	Días Real	Diferencia
Trabajo de Fin de Grado	88 días	101 días	-13 días
Planificación Inicial	03 días	02 días	01 días
Estudio estado de la cuestión	04 días	04 días	00 días
Estudio de soluciones similares	02 días	01 días	01 días
Estudio de elementos a utilizar	02 días	03 días	-01 días
Análisis	17 días	19 días	-02 días
Especificación de casos de uso	06 días	09 días	-03 días
Especificación de requisitos	11 días	10 días	01 días
Diseño	08 días	08 días	00 días
Diagramas de navegación	02 días	03 días	-01 días
Diagramas de la base de datos	02 días	02 días	00 días
Diseño de las pruebas	04 días	03 días	01 días
Preparación del sistema	00 días	05 días	-05 días
Instalación de Eclipse	00 días	01 días	-01 días
Configuración de Android Developer Tools	00 días	01 días	-01 días
Configuración de jQuery + jQuery Mobile	00 días	01 días	-01 días
Configuración de Phonegap	00 días	01 días	-01 días
Aprendizaje de jQuery y jQuery Mobile	00 días	01 días	-01 días



Fase	Días P.Inicial	Días Real	Diferencia
Aprendizaje de Phoneygap	00 días	02 días	-02 días
Compilación de HolaMundo en Phoneygap	00 días	01 días	-01 días
Implementación	43 días	51 días	-08 días
Pruebas	04 días	02 días	02 días
Documentación	09 días	10 días	-01 días

Tabla 95 - Comparativa entre la planificación inicial y el desarrollo real

Como se puede observar, ha habido una subestimación total de trece días. Algunos de los motivos que han podido originar esta desviación son: falta de experiencia previa, ya que se trata del primer proyecto realizado, subestimación de algunas tareas y sobrestimación de otras, y lo más importante, incluir nuevas tareas que no habían sido consideradas. Gracias a esta información, en el futuro se podrán afinar más las tareas de planificación, llegando un momento en el que la estimación sea una aproximación muy real de lo que finalmente ocurra.



5 Medios técnicos empleados para el proyecto

En este capítulo se muestra un breve resumen las herramientas que han sido utilizadas durante la realización del Trabajo de Fin de Grado. Se dividirán en herramientas *software* y herramientas *hardware*.

En la tabla que hay a continuación, se puede observar una relación de las diferentes herramientas *software* utilizadas y una breve descripción de las mismas.

Herramienta	Descripción
Eclipse 4.2.1 + Android Developer Tools v22.2.1	Entorno de desarrollo utilizado para la implementación de la aplicación.
Gantt Project 2.6.6 (build 1715)	Aplicación utilizada para la realización de la planificación y gestión del seguimiento del proyecto.
jQuery 1.9.2	<i>Framework</i> utilizado para la gestión de documentos HTML, eventos, etc.
jQuery Mobile 1.3.2	<i>Framework</i> utilizado para la realización de la interfaz de la aplicación.
Mac OSX 10.8.3	Sistema operativo sobre el que se han desarrollado las tareas relacionadas con el proyecto, tanto de desarrollo como de documentación.
Microsoft Office 2007	Suite ofimática que ha sido utilizada para el desarrollo de la documentación del Trabajo de Fin de Grado.
Phonegap 2.9.0	<i>Framework</i> utilizado para la realización de la aplicación móvil.
Safari 6.0.3	Navegador web utilizado durante el desarrollo del proyecto para la búsqueda de información, documentación, API, etc.
Star UML 5.0.2	Aplicación utilizada para la realización de los diagramas de casos de uso.

Tabla 96 - Medios técnicos empleados para el proyecto (software)



Análisis, diseño e implementación de un sistema para ayudar en el control de inventario.
Capítulo V: Medios técnicos empleados para el proyecto.

Además, en la siguiente tabla, se pueden observar también las herramientas *hardware* que han sido utilizadas.

Herramienta	Descripción
Macbook Pro 13" retina 2,4 Ghz, 8Gb de RAM, 128Gb SSD	Ordenador portátil que ha sido utilizado para la realización del Trabajo de Fin de Grado.
Samsung Galaxy S3 Mini 4" 1,2Ghz 1Gb RAM	Móvil en el que ha sido desarrollada y depurada la aplicación.

Tabla 97 - Medios técnicos empleados para el proyecto (hardware)



6 Presupuesto

En este capítulo se realizará un análisis económico del proyecto, con el fin de detallar los costes asociados al mismo. Para ello se explicará la metodología utilizada para estimar los costes, y se expondrán el presupuesto inicial, el presupuesto a entregar al cliente y el coste real que finalmente ha tenido el proyecto. Por último, también se realizará un análisis de la desviación producida entre el presupuesto inicial y el coste real del proyecto.

6.1 Metodología de estimación de costes

Para realizar la estimación de costes se han tenido en cuenta costes directos e indirectos.

Los costes directos son aquellos que están directamente relacionados con el desarrollo del proyecto. Esto es: recursos humanos, herramientas *hardware*, herramientas *software*, fungibles, dietas y costes de viaje.

Los costes indirectos son aquellos que no están directamente relacionados con el desarrollo del proyecto, pero que son necesarios para posibilitar su desarrollo. Esto es: teléfono, conexión a internet, alquiler del inmueble en el que se realizará el proyecto y luz, entre otros. Para calcularlos, se han estimado como un 20% de los costes directos, tal y como sugiere la plantilla.

6.2 Presupuesto inicial

En este apartado se detallarán cada una de las estimaciones de costes para cada una de las áreas definidas anteriormente, así como la suma total de todos ellos. Éstos han sido calculados de la siguiente manera:

- Recursos humanos: costes asociados a los sueldos de los empleados.
- Herramientas *hardware*: precio de venta al público del *hardware* que ha sido utilizado durante la realización del proyecto.
- Herramientas *software*: precio de venta al público y licencias de estudiante gratuitas del *software* que ha sido utilizado durante la realización del proyecto.
- Fungibles: costes asociados a material de oficina, etc. Se refiere a todos aquellos elementos que se consumen con el uso.
- Dietas: cantidades acordadas previamente para cubrir gastos de comidas, alojamiento, etc.
- Costes de viaje: cantidad fijada por kilómetro, peajes, etc.

El 21% de IVA está incluido únicamente en la suma del coste total del proyecto, dato que se especifica explícitamente en cada uno de los cálculos.

6.2.1 Costes asociados a Recursos Humanos

En esta sección se detallan los costes imputados a los empleados que participan en cualquier fase del proyecto. En este caso hay un único empleado, pero los sueldos estarán diferenciados en virtud de la función que se desempeñe en cada fase del proyecto.

Apellidos, Nombre	Categoría	Dedicación horas	Dedicación (hombres mes) ¹	Coste hombre/mes (sin IVA) ²	Coste (sin IVA)
Gómez González, Sergio	Jefe de Proyecto ³	35 h	0,268190476	4000 €	1073 €
Gómez González, Sergio	Analista	96 h	0,731428571	2800 €	2048 €
Gómez González, Sergio	Diseñador	32 h	0,243809524	2000 €	488 €
Gómez González, Sergio	Programador	172 h	1,31047619	1400 €	1835 €
Gómez González, Sergio	Pruebas	16 h	0,121904762	1400 €	171 €
Gómez González, Sergio	Documentación	36 h	0,274285714	1200 €	329 €
Total		352 h	2,950095238	12800 €	5943 €

Tabla 98 - Presupuesto inicial: Costes asociados a Recursos Humanos

¹ 1 hombre mes = 131,25 horas. Máximo anual de dedicación de 12 hombres mes (1575 horas).
Máximo anual para PDI de la Universidad Carlos III de Madrid de 8,8 hombres mes (1155 horas).



²Los salarios han sido sacados de la página Plan de Carrera de Infojobs [26]

³Se ha considerado que el Jefe de Proyecto imputa un 10% de las horas totales del proyecto (352h), pero al tratarse de labores de coordinación y supervisión, no se han tenido en cuenta en la suma total de horas.

*En el caso de que alguna tarea hubiera sido subcontratada, debería aparecer reflejado en este apartado.

6.2.2 Costes asociados a herramientas hardware

En esta sección se detallan los costes imputados a los equipos utilizados durante la realización del proyecto. Estos costes se imputan en función del precio de venta al público, el porcentaje de uso dedicado y el periodo de depreciación. El periodo de depreciación se ha definido en 36 meses. En el caso del teléfono móvil, el periodo de depreciación es de 12 meses, que es aproximadamente el tiempo transcurrido entre lanzamientos de nuevas versiones de una misma línea de terminales. En la siguiente tabla se puede ver un resumen de estos equipos y de los costes asociados a los mismos.

Descripción	Coste (sin IVA)	Dedicación	Periodo de depreciación	Coste imputable
Portátil Macbook Pro Retina 13 -inch 2012	1.098,00 €	4 meses	36 meses	122,00 €
Teléfono Samsung Galaxy S3 Mini	220,41 €	4 meses	12 meses	73,47 €
Impresora HP Deskjet 2540	59,00 €	4 meses	36 meses	6,56 €
Coste total imputable				202,03 €

Tabla 99 - Presupuesto inicial: Costes asociados a herramientas hardware

6.2.3 Costes asociados a herramientas software

En esta sección se detallan los costes imputados al *software* utilizado durante la realización del proyecto. Estos costes se imputan del mismo modo que en la sección anterior. En la siguiente tabla se puede ver un resumen de dicho *software* y de los costes asociados al mismo.



Descripción	Coste (sin IVA)	Dedicación	Periodo de depreciación	Coste imputable
Mac OSX 10.8 ¹	0,00 €	4 meses	36 meses	0,00 €
Microsoft Office 2007 ²	0,00 €	4 meses	36 meses	0,00 €
Gantt Project 2.6.6 (build 1715) ³	0,00 €	4 meses	36 meses	0,00 €
Star UML 5.0.2 ³	0,00 €	4 meses	36 meses	0,00 €
Skype ³	0,00 €	4 meses	36 meses	0,00 €
Coste total imputable				0,00 €

Tabla 100 - Presupuesto inicial: Costes asociados a herramientas software

¹Instalado en el portátil en el momento de la compra.

²Licencia gratuita de estudiante.

³Software gratuito.

6.2.4 Costes asociados a fungibles

En esta sección se detallan los costes asociados a fungibles. En el caso de este proyecto, dichos costes se reducen a material de oficina como por ejemplo cartuchos de tinta para la impresora, folios, carpetas, bolígrafos, *whiteboard*, etc. En la siguiente tabla se puede ver un resumen de estos costes:

Descripción	Coste unitario (sin IVA)	Cantidad	Coste Total
Cartuchos de impresora	23,99 €	2	47,98 €
Folios	3,50 €	2	7,00 €
Bolígrafos	0,60 €	20	12,00 €
Whiteboard	20,00 €	1	20,00 €
Otros	40,00 €	1	40,00 €
Total			126,98 €

Tabla 101 - Presupuesto inicial: Costes asociados a fungibles

6.2.5 Costes asociados a dietas y costes de viaje

En esta sección se detallan los costes imputados a dietas y costes derivados de los viajes realizados durante el desarrollo del proyecto.



Análisis, diseño e implementación de un sistema para ayudar en el control de inventario.
Capítulo VI: Presupuesto.

Se ha asignado un coste diario de 8,50€ en concepto de dietas. No ha habido costes relacionados con alojamiento.

Se considera cliente a D. Alejandro Calderón Mateos, tutor del trabajo de fin de grado, cuyo despacho está en el campus de Leganés de la UC3M. Se establece una reunión semanal con el fin de tener un *feedback* del cliente. La distancia de cada trayecto (ida y vuelta) es de 20,6 km con un coste acordado de 0,121€/km. Con un total de 17 reuniones, hace un total de 42,3742€.

En la siguiente tabla se puede ver un resumen de estos costes:

Descripción	Coste unitario (sin IVA)	Cantidad	Coste Total
Costes desplazamiento	0,1210 €/km	350,2 km	42,37 €
Dietas	8,50 €/día	88 días	748,00 €
Total			790,37 €

Tabla 102 - Presupuesto inicial: Costes asociados a dietas y costes de viaje

6.2.6 Total costes directos

En esta sección se detalla el total de los costes directos asociados al proyecto. Para su cálculo, se suman todos los costes parciales asociados a cada uno de los conceptos detallados en los apartados anteriores:

- Recursos humanos.
- Herramientas *hardware*.
- Herramientas *software*.
- Fungibles.
- Dietas y costes de viaje.

En la siguiente tabla se puede ver un resumen de los costes directos:

Concepto	Coste
Costes asociados a Recursos Humanos	5.942,86 €
Costes asociados a herramientas hardware	202,03 €
Costes asociados a herramientas software	0,00 €
Costes asociados a fungibles	126,98 €
Costes asociados a dietas y costes de viaje	790,37 €
Total Costes Directos	7.062,24 €

Tabla 103 - Presupuesto inicial: Total costes directos



6.2.7 Total costes indirectos

Como se ha especificado anteriormente, los costes indirectos se han calculado como un 20% de los costes directos que han sido indicados en el anterior apartado.

En la siguiente tabla se puede ver un resumen de los costes indirectos:

Total Costes Directos	Fracción aplicada	Total Costes Indirectos
7.062,24 €	20%	1.412,45 €

Tabla 104 - Presupuesto inicial: Total costes indirectos

6.2.8 Estimación inicial de costes

En esta sección se detalla el total de los costes estimados para el proyecto. Para su cálculo, se suman los costes directos y los costes indirectos calculados en los apartados anteriores. A dicha suma le será añadido el IVA, y se obtendrá una estimación inicial de costes.

En la siguiente tabla se puede ver un resumen de la estimación de costes inicial:

Concepto	Coste
Costes asociados a Recursos Humanos	5.942,86 €
Costes asociados a herramientas hardware	202,03 €
Costes asociados a herramientas software	0,00 €
Costes asociados a fungibles	126,98 €
Costes asociados a dietas y costes de viaje	790,37 €
Total Costes Directos	7.062,24 €
Total Costes Indirectos	1.412,45 €
Total costes sin IVA	8.474,68 €
IVA (21%)	1.779,68 €
Total	10.254,37 €

Tabla 105 - Presupuesto inicial: Estimación inicial de costes



6.2.9 Presupuesto a entregar al cliente

En esta sección se detalla el presupuesto que se entregará al cliente. En él, se suma a la estimación inicial de costes total vista en la sección anterior un porcentaje de riesgo con el que poder hacer frente a problemas e imprevistos. Se ha decidido que dicho porcentaje sea de un 10%. Dicho valor se ha definido de acuerdo a lo visto durante la realización de la carrera. Al coste resultante se le añadirá adicionalmente los beneficios que se esperan conseguir por la realización del proyecto. Estos beneficios estarán expresados como un porcentaje de la estimación de costes total (riesgo incluido). Se ha decidido que este porcentaje sea de un 15%. Dicho valor se ha definido de acuerdo a lo visto durante la realización de la carrera.

La aplicación será desarrollada para la empresa que paga el presupuesto. Dicha empresa, puede ofertar la aplicación a terceras empresas, pudiéndose adaptar a las necesidades concretas de cada cliente. Por dichas adaptaciones, los desarrolladores percibirán un 25% de cada venta que se realice. Además, se pagará un mantenimiento de 25€/al año, que los desarrolladores percibirán en su totalidad. En la siguiente tabla se puede ver un resumen del presupuesto que se le entregará al cliente:

Concepto	Coste
Costes asociados a Recursos Humanos	5.942,86 €
Costes asociados a herramientas hardware	202,03 €
Costes asociados a herramientas software	0,00 €
Costes asociados a fungibles	126,98 €
Costes asociados a dietas y costes de viaje	790,37 €
Total Costes Directos	7.062,24 €
Total Costes Indirectos	1.412,45 €
Total costes sin IVA antes de riesgo	8.474,68 €
Riesgo (10%)	847,47 €
Total costes sin IVA antes de beneficio	9.322,15 €
Beneficios (15%)	1.398,32 €
Total Costes sin IVA	10.720,48 €
IVA (21%)	2.251,30 €
Total	12.971,78 €

Tabla 106 - Presupuesto inicial: Presupuesto a entregar al cliente



6.3 Coste final del proyecto

En este apartado se detalla el coste que finalmente ha tenido el proyecto. Al igual que en la estimación de costes, se detallarán los costes para cada una de las áreas en las que se ha dividido el presupuesto. Han sido calculados de la misma manera que el presupuesto inicial. Se han omitido las explicaciones de cada una de las áreas. Las mismas se pueden encontrar en el apartado Presupuesto inicial.

6.3.1 Costes asociados a Recursos Humanos

Como se puede observar en la siguiente tabla, el aumento del número de horas dedicadas para la realización del presente Trabajo de Fin de Grado, repercute directamente en los costes asociados a recursos humanos, aumentando los mismos en 750€.

Apellidos, Nombre	Categoría	Dedicación horas	Dedicación (hombres mes) ¹	Coste hombre/mes (sin IVA) ²	Coste (sin IVA)
Gómez González, Sergio	Jefe de Proyecto ³	40 h	0,307809524	4000 €	1231 €
Gómez González, Sergio	Analista	100 h	0,761904762	2800 €	2133 €
Gómez González, Sergio	Diseñador	32 h	0,243809524	2000 €	488 €
Gómez González, Sergio	Programador	224 h	1,706666667	1400 €	2389 €
Gómez González, Sergio	Pruebas	8 h	0,060952381	1400 €	85 €
Gómez González, Sergio	Documentación	40 h	0,304761905	1200 €	366 €
Total		404 h	3,385904762	12800 €	6693 €

Tabla 107 - Presupuesto inicial: Costes asociados a Recursos Humanos

¹1 hombre mes = 131,25 horas. Máximo anual de dedicación de 12 hombres mes (1575 horas).

Máximo anual para PDI de la Universidad Carlos III de Madrid de 8,8 hombres mes (1155 horas).

²Los salarios han sido sacados de la página Plan de Carrera de Infojobs [27]

³Se ha considerado que el Jefe de Proyecto imputa un 10% de las horas totales del proyecto (404h), pero al tratarse de labores de coordinación y supervisión, no se han tenido en cuenta en la suma total de horas.

*En el caso de que alguna tarea hubiera sido subcontratada, debería aparecer reflejado en este apartado.



6.3.2 Costes asociados a herramientas hardware

En este caso no ha habido cambios respecto a los costes que se habían estimado inicialmente, ya que se trata de un PVP que no ha variado desde el momento en el que se estimaron los costes hasta que se compró el *hardware* necesario.

6.3.3 Costes asociados a herramientas software

En este caso no ha habido cambios respecto a los costes que se habían estimado inicialmente, ya que todo el *software* es gratuito, o bien se ha conseguido gracias a licencias gratuitas de estudiante.

6.3.4 Costes asociados a fungibles

En este caso la variación de los costes asociados a fungibles respecto a los estimados inicialmente, es prácticamente despreciable.

Descripción	Coste unitario (sin IVA)	Cantidad	Coste Total
Cartuchos de impresora	23,99 €	3	71,97 €
Folios	3,50 €	2	7,00 €
Bolígrafos	0,60 €	15	9,00 €
Whiteboard	20,00 €	1	20,00 €
Otros	46,54 €	1	46,54 €
Total			154,51 €

Tabla 108 - Coste final del proyecto: Costes asociados a fungibles

6.3.5 Costes asociados a dietas y costes de viaje

Los costes asociados a dietas y costes de viaje se han visto alterados por los siguientes motivos: por un lado, se había estimado realizar 17 reuniones con el cliente. Por motivos de disponibilidad, las reuniones presenciales han sido finalmente menos. Si bien, se han realizado reuniones de acuerdo a la planificación mediante *Skype*, las reuniones que han requerido desplazamiento, y por lo tanto, llevan un coste asociado han sido sólo 5.



Por otro lado, el número de días que se ha tardado en desarrollar el proyecto, ha aumentado de 88 a 101 días, por lo que el coste asociado a las dietas ha aumentado considerablemente.

Descripción	Coste unitario (sin IVA)	Cantidad	Coste Total
Costes desplazamiento	0,1210 €/km	103,0 km	12,46 €
Dietas	8,50 €/día	101 días	858,50 €
Total			870,96 €

Tabla 109 - Coste final del proyecto: Costes asociados a dietas y costes de viaje

6.3.6 Total costes directos

En esta sección se detalla el total de los costes directos reales asociados al proyecto. Para su cálculo, se suman todos los costes parciales asociados a cada uno de los conceptos detallados en los apartados anteriores:

- Recursos humanos.
- Herramientas *hardware*.
- Herramientas *software*.
- Fungibles.
- Dietas y costes de viaje.

En la siguiente tabla se puede ver un resumen de los costes directos:

Concepto	Coste
Costes asociados a Recursos Humanos	6.692,57 €
Costes asociados a herramientas hardware	202,03 €
Costes asociados a herramientas software	0,00 €
Costes asociados a fungibles	154,51 €
Costes asociados a dietas y costes de viaje	870,96 €
Total Costes Directos	7.920,07 €

Tabla 110 - Coste final del proyecto: Total costes directos



6.3.7 Total costes indirectos

Como ya se ha especificado anteriormente, los costes indirectos se han calculado como un 20% de los costes directos indicados en el anterior apartado.

En la siguiente tabla se puede ver un resumen de los costes indirectos:

Total Costes Directos	Fracción aplicada	Total Costes Indirectos
7.920,07 €	20%	1.584,01 €

Tabla 111 - Coste final del proyecto: Total costes indirectos

6.3.8 Coste real del proyecto

En esta sección se detalla el total de los costes reales del proyecto. Para su cálculo, se suman los costes directos y los costes indirectos calculados en los apartados anteriores. A dicha suma le será añadido el IVA, y se obtendrá el coste real del proyecto.

Concepto	Coste
Costes asociados a Recursos Humanos	6.692,57 €
Costes asociados a herramientas hardware	202,03 €
Costes asociados a herramientas software	0,00 €
Costes asociados a fungibles	154,51 €
Costes asociados a dietas y costes de viaje	870,96 €
Total Costes Directos	7.920,07 €
Total Costes Indirectos	1.584,01 €
Total costes sin IVA	9.504,08 €
IVA (21%)	1.995,86 €
Total	11.499,94 €

Tabla 112 - Coste final del proyecto: Coste real del proyecto



6.4 Análisis de la desviación

En la siguiente tabla se puede observar una comparación entre el coste que inicialmente se había estimado, y el coste que finalmente ha tenido el proyecto.

Concepto	Coste Planificado	Coste Real	Diferencia
Costes asociados a Recursos Humanos	5.942,86 €	6.692,57 €	749,71 €
Costes asociados a herramientas hardware	202,03 €	202,03 €	0,00 €
Costes asociados a herramientas software	0,00 €	0,00 €	0,00 €
Costes asociados a fungibles	126,98 €	154,51 €	27,53 €
Costes asociados a dietas y costes de viaje	790,37 €	870,96 €	80,59 €
Total Costes Directos	7.062,24 €	7.920,07 €	857,83 €
Total Costes Indirectos	1.412,45 €	1.584,01 €	171,57 €
Total costes sin IVA	8.474,68 €	9.504,08 €	1.029,40 €
IVA (21%)	1.779,68 €	1.995,86 €	216,17 €
Total	10.254,37 €	11.499,94 €	1.245,57 €

Tabla 113 - Análisis de la desviación de costes

Como se puede observar, los costes reales son superiores a los estimados, por un total de 1.245,57€. En el presupuesto a entregar al cliente, de 12.971,78 €, se habían incluido 1025,43€ (IVA incluido) en concepto de riesgo, y 1691,97 € (IVA incluido) en concepto de beneficios. Para cubrir el sobrecoste se utiliza el importe calculado como riesgo. Dado que en este caso no se llega a cubrir el total de dicho sobrecoste, habría que utilizar 221,14 € (IVA incluido) adicionales del beneficio, lo que deja unos beneficios finales de 1470,83 €, un 13% menos de lo esperado.



7 Conclusiones y trabajos futuros

7.1 Conclusiones

En este apartado se comentarán las conclusiones que se han extraído tras la realización del Trabajo de Fin de Grado. Éstas han sido divididas en los siguientes tipos de conclusiones:

- Conclusiones relativas al sistema desarrollado: se hará un repaso de los objetivos que fueron planteados en el apartado de Objetivos y si han sido logrados o no.
- Conclusiones relativas a la planificación y el presupuesto: se hará un repaso del proceso que se ha seguido para la realización del Trabajo de Fin de Grado, comentando que problemas se han dado en la planificación, y que se ha aprendido de ellos.
- Conclusiones personales: se hará un repaso de los conocimientos aplicados en el Trabajo de Fin de Grado que han sido adquiridos durante la carrera, las asignaturas en las que se adquirieron dichos conocimientos, así como los conocimientos nuevos que se han aprendido durante la realización del Trabajo de Fin de Grado.

7.1.1 Conclusiones relativas al sistema desarrollado

En esta sección se repasará la consecución de los objetivos que han sido establecidos al inicio del presente documento.

- Permitir a los usuarios actuar *in situ* sin necesidad de estar delante de un ordenador.
 - ✓ Creando una aplicación para *smartphone* este objetivo queda cumplido, ya que los usuarios pueden actuar desde cualquier lugar sin necesidad de disponer de un ordenador, y pueden actuar en el preciso instante en el que el cambio tiene lugar.
- Diseño intuitivo.
 - ✓ El diseño no cuenta con más elementos de los imprescindibles para poder controlar el inventario, dejando a un lado la sobreinformación y el ofrecer muchas opciones, dos características que hacen que los usuarios acaben perdidos mientras navegan por la aplicación. Por tanto, se puede decir que este objetivo se ha cumplido.
- Agilizar el control del inventario informático en una empresa y/o entidad.
 - ✓ Gracias al cumplimiento de los dos objetivos anteriores, se permite agilizar el control del inventario, por lo que el objetivo se puede considerar cumplido.



- Mantener en todo momento una representación fiel de la realidad, y mostrar los datos de tal forma que facilite la toma de decisiones.
 - ✓ Consiguiendo cumplir todos los objetivos anteriores, se consigue mantener una representación fiel de la realidad, ya que los cambios pueden hacerse al instante, evitando posibles olvidos y despistes; se evitan posibles errores, al tener un diseño intuitivo y sólo con aquella información relevante, evitando saturar de información al usuario, y teniendo botones auto explicativos que evitan posibles confusiones al utilizarlos.

7.1.2 Conclusiones relativas a la planificación y el presupuesto

En esta sección se repasarán los problemas que se han dado en la planificación del proyecto, así como un breve estudio de cómo se hubiera desarrollado el dicho proyecto (planificación y costes), si se hubiera podido trabajar en él 8 horas diarias, es decir, una jornada laboral completa.

7.1.2.1 Problemas de planificación

Como ya se ha explicado en el capítulo de Planificación del proyecto, se ha producido una desviación entre la planificación que inicialmente se había estimado y el cómo se ha desarrollado finalmente el proyecto. Es fácil haber cometido errores de planificación, ya que se trata del primer proyecto que se hace, y es difícil estimar la duración de algunas tareas, ya que no hay conocimiento previo. Para futuros proyectos, y teniendo en cuenta los datos del presente proyecto, se podrá hacer una estimación que se ajuste más a la realidad, y se tendrán en cuenta tareas que inicialmente se habían obviado, como por ejemplo, la preparación del sistema y el entorno en el que se va a desarrollar el proyecto.

7.1.2.2 Escenario alternativo en el desarrollo del proyecto

Como se ha podido ver en el apartado de Planificación del Proyecto, el Trabajo de Fin de Grado se ha realizado con una dedicación de 4 horas al día, debido a que se ha tenido que compaginar con una jornada laboral de 5 horas diarias. En este apartado se puede observar cómo se hubiera desarrollado el proyecto en términos de planificación y costes, si el mismo se hubiera podido realizar en circunstancias normales, es decir, con una dedicación de 8 horas diarias, como si de una jornada laboral se tratase.



7.1.2.2.1 *Planificación Alternativa*

En esta sección se detallará la planificación que hubiera tenido el proyecto en el caso de haber dedicado 8 horas diarias a su desarrollo.

Fase	Días	Horas	% Total
Trabajo de Fin de Grado	51 días	404 horas	100,00%
Planificación Inicial	01 días	08 horas	1,98%
Estudio estado de la cuestión	02 días	16 horas	3,96%
Estudio de soluciones similares	01 días	04 horas	0,99%
Estudio de elementos a utilizar	02 días	12 horas	2,97%
Análisis	05 días	40 horas	9,90%
Especificación de casos de uso	05 días	36 horas	8,91%
Especificación de requisitos	05 días	40 horas	9,90%
Diseño	08 días	64 horas	15,84%
Diagramas de navegación	02 días	16 horas	3,96%
Diagramas de la base de datos	02 días	12 horas	2,97%
Diseño de las pruebas	02 días	12 horas	2,97%
Preparación del sistema	03 días	20 horas	4,95%
Instalación de Eclipse	01 días	02 horas	0,50%
Configuración de Android Developer Tools	01 días	02 horas	0,50%
Configuración de jQuery + jQuery Mobile	01 días	02 horas	0,50%
Configuración de Phonegap	01 días	02 horas	0,50%
Aprendizaje de jQuery y jQuery Mobile	01 días	04 horas	0,99%
Aprendizaje de Phonegap	01 días	06 horas	1,49%
Compilación de HolaMundo en Phonegap	01 días	02 horas	0,50%
Implementación	26 días	208 horas	51,49%
Pruebas	01 días	08 horas	1,98%
Documentación	05 días	40 horas	9,90%

Tabla 114 - Planificación alternativa



Análisis, diseño e implementación de un sistema para ayudar en el control de inventario.
Capítulo VII: Conclusiones y trabajos futuros.

Como era de esperar, pudiendo dedicar el doble de horas al día, la duración del proyecto se reduciría a la mitad.

7.1.2.2.2 Costes alternativos del proyecto

En este apartado se detallarán los costes que hubiera tenido el proyecto en este escenario alternativo. No se va a entrar en el mismo nivel de detalle que en el apartado de Coste final del proyecto, si no que se va a mostrar la tabla de resumen final, comparando los costes alternativos con los que realmente ha tenido el proyecto, y sobre ella se harán los comentarios oportunos.

Concepto	Coste Real	Coste Alternativo	Diferencia
Costes asociados a Recursos Humanos	6.692,57 €	6.692,57 €	0,00 €
Costes asociados a herramientas hardware	202,03 €	101,01 €	101,01 €
Costes asociados a herramientas software	0,00 €	0,00 €	0,00 €
Costes asociados a fungibles	154,51 €	154,51 €	0,00 €
Costes asociados a dietas y costes de viaje	870,96 €	445,96 €	425,00 €
Total Costes Directos	7.920,07 €	7.394,06 €	526,01 €
Total Costes Indirectos	1.584,01 €	1.478,81 €	105,20 €
Total costes sin IVA	9.504,08 €	8.872,87 €	631,22 €
IVA (21%)	1.995,86 €	1.863,30 €	132,56 €
Total	11.499,94 €	10.736,17 €	763,77 €

Tabla 115 - Costes alternativos del proyecto

Los costes asociados a Recursos Humanos no han sufrido cambios, debido a que el número de horas dedicadas por cada rol es el mismo, aunque el número de días necesarios para llegar a dedicarlas se haya reducido a la mitad.

Los costes asociados a herramientas *hardware* han sufrido cambios, ya que los meses de dedicación se han reducido a la mitad, y por tanto, el coste imputable también.

Los costes asociados a herramientas *software* no han sufrido cambios, ya que aunque la dedicación también se ha reducido a la mitad como en el caso anterior, ningún *software* de los que se ha utilizado tenía coste asociado. En caso de que hubiera habido costes imputables a herramientas *software*, también se hubiera reducido a la mitad por el mismo motivo que los costes asociados a herramientas *hardware*.



Análisis, diseño e implementación de un sistema para ayudar en el control de inventario.
Capítulo VII: Conclusiones y trabajos futuros.

Los costes asociados a fungibles se mantienen idénticos, ya que son elementos que son necesarios igualmente, independientemente de la duración del proyecto.

Los costes asociados a dietas y costes de viaje se han visto reducidos a prácticamente la mitad. El motivo es que los costes de dieta se reducen a la mitad, ya que se aplican la mitad de días. Se ha considerado que los costes de viaje se mantienen, al mantener el mismo número de reuniones presenciales.

Todo esto repercute directamente en un descenso de los costes directos, y por consecuencia, de los indirectos. El coste final del proyecto sería de 10.736,17€, es decir, 763,77€ menos de lo que realmente ha costado llevarlo a cabo.

Como conclusión se extrae que habiendo dedicado el doble de horas al día, se podría haber reducido costes en un 6,7% aproximadamente, lo que conllevaría poder ofrecer un precio más competitivo al cliente, o bien, aumentar el porcentaje de beneficios hasta un 24% aproximadamente.

7.1.3 Conclusiones personales

Durante la realización de la carrera he adquirido conocimientos, muchos de los cuales me han sido útiles para la realización del Trabajo de Fin de Grado. Cabe destacar las siguientes asignaturas que han aportado dichos conocimientos:

- **Programación** (1er curso, 1er cuatrimestre): es la asignatura de contacto con el mundo de la programación, y la base que cimienta todos los conocimientos que se han adquirido respecto a esta competencia. El lenguaje que se vio en dicha asignatura fue Java.
- **Programación Orientada a Objetos** (1er curso, 2º cuatrimestre): primera asignatura en la que se tuvo contacto con la programación orientada a objetos. Nos permitió entender el concepto de objeto en programación, y cómo trabajar con ellos.
- **Ficheros y Bases de Datos** (2º curso, 2º cuatrimestre): en esta asignatura se estableció contacto con los sistemas de ficheros y las bases de datos, aprendiendo su funcionamiento, y sirviendo como toma de contacto con SQL, lenguaje base que ha sido utilizado para el manejo de la base de datos en la aplicación.
- **Seguridad en tecnologías de la información** (2º curso, 2º cuatrimestre): en esta asignatura se estableció contacto con los temas de seguridad informática (por ejemplo, tipos de cifrado) y por qué es tan importante este aspecto en el desarrollo de *software*.
- **Interfaces de Usuario** (2º curso, 2º cuatrimestre): en esta asignatura se adquirieron conocimientos de HTML, CSS, Javascript y algo de jQuery, que han sido necesarios para la realización del Trabajo de Fin de Grado.
- **Procesos del Desarrollo del Software** (3er curso, 1er cuatrimestre): en esta asignatura se estableció por primera vez contacto con los procesos relacionados con la gestión del proyecto.



Análisis, diseño e implementación de un sistema para ayudar en el control de inventario.
Capítulo VII: Conclusiones y trabajos futuros.

- **Diseño y Administración de Bases de Datos** (4º curso, 1er cuatrimestre): en esta asignatura he ampliado mis conocimientos sobre bases de datos, que me han ayudado a modelar mejor la base de datos que soporta la aplicación.
- **Tecnologías Informáticas para la Web** (4º curso, 1er cuatrimestre): en esta asignatura se han adquirido conocimientos sobre aplicaciones web, y aunque durante su desarrollo se usaba JEE, me ha servido para manejar con mayor facilidad algunos conceptos, como por ejemplo la interacción del HTML (JSP en JEE) con la base de datos.
- **Dirección de Proyectos de Desarrollo de Software** (4º curso, 1er cuatrimestre): en esta asignatura se aprendieron conocimientos sobre las metodologías pesadas, concretamente Métrica 3.
- **Metodología del Desarrollo Visual** (4º curso, 1er cuatrimestre): en esta asignatura, he aprendido el ciclo completo que ha de seguirse para desarrollar *software* siguiendo una metodología ágil.
- **Técnicas Ágiles de Desarrollo Software** (4º curso, 2º cuatrimestre): en esta asignatura he afianzado los conocimientos que tenía de las metodologías ágiles. En ella se desarrolló un *software* mediante un proceso iterativo e incremental, y en cada una de las iteraciones se entregaba *software* totalmente funcional. Aprendí la importancia de la comunicación entre los miembros del equipo en este tipo de metodologías.

Además, durante la realización del Trabajo de Fin de Grado he adquirido por cuenta propia otros conocimientos nuevos:

- Programar con distintos lenguajes y entornos: se ha aprendido a programar utilizando diferentes lenguajes que habían sido poco o nada utilizados durante la carrera, como por ejemplo:
 - jQuery: aunque durante la realización de la asignatura Interfaces de Usuario se utilizó algo de jQuery, ha sido durante la realización del Trabajo de Fin de Grado cuando he afianzado de verdad estos conocimientos.
 - jQuery Mobile: He aprendido a desarrollar interfaces para aplicaciones móviles gracias al *framework* jQuery Mobile.

Aparte de los lenguajes, se ha aprendido a programar para entornos móviles, gracias a la herramienta Phonegap:

- Phonegap: he aprendido un nuevo entorno para realizar aplicaciones móviles. Me parece una opción con mucho potencial, ya que al basar las aplicaciones en vistas web permite un alto grado de portabilidad, sirviendo el mismo código para diferentes SO: Android, iOS, Windows Phone, etc.



7.2 Trabajos futuros

Como posibles mejoras del prototipo de aplicación realizada durante el Trabajo de Fin de Grado, cabe destacar las siguientes:

- Actualización a la última versión de Phonegap.
- Actualización a la última versión de jQuery y jQuery Mobile.
- Importación y exportación de listados a CSV.
- Exportación de listados a PDF.
- Sincronización de diferentes dispositivos.
- Multilenguaje.
- Inventario de otros elementos informáticos.
- Mensajería interna.
- Opciones de accesibilidad.
- Diferentes roles para los usuarios.
- Estadísticas.
- Otros ámbitos de la aplicación.
- Mejora continua.

7.2.1 Actualización a la última versión de Phonegap

Phonegap va evolucionando constantemente. Si bien para el proyecto se ha utilizado la versión 2.9, actualmente el *framework* se encuentra en la versión 3.4 (11/06/14). Para el futuro, sería interesante la actualización del mismo, para agregar a la aplicación todas las novedades, y para poder crear nueva funcionalidad que se vaya incluyendo en las nuevas actualizaciones.

7.2.2 Actualización a la última versión de jQuery y jQuery Mobile

Del mismo modo que Phonegap, jQuery también va evolucionando constantemente. La versión que se ha utilizado para el desarrollo del Trabajo de Fin de Grado es la 1.9.2. Actualmente, la última versión disponible es la 1.11.1 (11/06/14) o bien la 2.1.1 (11/06/14), que no es compatible con Internet Explorer 6, 7 u 8.

Al igual que pasa con jQuery, jQuery Mobile sigue evolucionando. La versión que se ha utilizado para el desarrollo del Trabajo de Fin de Grado es la 1.3.2. Actualmente, la última versión disponible es la 1.4.2 (11/06/14). En el futuro sería interesante actualizar a la última versión disponible en ese momento.



7.2.3 Importación y exportación de listados a CSV

En una aplicación de este tipo sería muy recomendable tener la posibilidad de exportar los listados generados (por ejemplo, de máquinas o de usuarios) a un formato externo, para poder consultarlos desde el exterior de la aplicación. Del mismo modo sería interesante poder trabajar con estos archivos externos, pudiendo realizar modificaciones en los mismos, y poder importar dicho archivo para que los cambios fueran automáticamente actualizados en la base de datos.

7.2.4 Exportación de listados a PDF

También sería muy recomendable disponer de la opción de exportar los listados a PDF, de modo que puedan ser impresos de un modo más legible y ordenado que mediante listados CSV. Dichos listados serían simplemente informativos, ya que no se podría volver a alimentar la base de datos con los mismos.

7.2.5 Sincronización entre diferentes dispositivos

Dado que la aplicación está pensada para que varios usuarios puedan gestionar un mismo inventario, sería muy interesante permitir que la base de datos se pudiera sincronizar en diferentes dispositivos automáticamente al entrar en la aplicación. Dado que la opción más sencilla (*Datastores* de *Dropbox Sync*) no se ha podido llevar a cabo, queda como trabajo futuro, lo que dotaría a la aplicación de más valor añadido.

7.2.6 Multilenguaje

Otro de los trabajos futuros que se podrían llevar a cabo para mejorar la aplicación sería el permitir al usuario elegir el idioma de la interfaz: español, inglés o francés. Esto permitiría llegar a un mayor número de gente, y que usuarios de multitud de nacionalidades pudieran utilizar la aplicación sin el problema de la barrera del idioma.

7.2.7 Inventario de otros elementos informáticos

Aunque inicialmente la aplicación se centra en el inventario de ordenadores, se podría ampliar para poder inventariar otros elementos como por ejemplo monitores, teclados, ratones, etc. De esta forma se llevaría un control más exhaustivo de todos los elementos que conforman el parque informático.



7.2.8 Mensajería interna

Sería interesante incluir la opción de permitir enviar mensajes internos en la aplicación. Estos mensajes no se enviarían al correo electrónico del usuario, si no que aparecerían como notificación de la aplicación cuando un usuario accediera a la misma. Se podría incluir la opción de notificar por correo electrónico estos mensajes si el usuario lo desea.

7.2.9 Opciones de accesibilidad

En el futuro también sería interesante incluir un modo que permitiera a las personas con problemas visuales tener una mayor accesibilidad a la aplicación. Para conseguirlo, el usuario podría activar un modo de contraste máximo, o aumentar el tamaño de la fuente.

7.2.10 Diferentes roles para los usuarios

Como se ha comentado anteriormente, el prototipo de aplicación propuesto está pensado para un usuario único con acceso a toda la funcionalidad de la misma. Sería necesario en el futuro crear diferentes roles. Cada rol tendría permitido y prohibido hacer determinadas acciones en el sistema. Por ejemplo, un usuario normal no podría borrar tablas de la base de datos, ni el log.

7.2.11 Estadísticas

Sería interesante implementar una sección de estadísticas, donde pudiera verse de un simple vistazo datos relevantes para el control del inventario, por ejemplo: número de máquinas rotas, número de máquinas fuera de la oficina, máquinas en stock, etc.

7.2.12 Otros ámbitos de la aplicación

Aunque la aplicación realizada durante el Trabajo de Fin de Grado está orientada al control de inventario informático (concretamente a los ordenadores), con unas ligeras modificaciones también podría ser utilizada para el control de inventario de cualquier tipo: biblioteca, videoclub, tienda...El funcionamiento sería similar, y el propósito el mismo: permitir controlar un inventario, de un modo simple, eficiente y portátil.

7.2.13 Mejora continua

Se tendría muy en cuenta el *feedback* de los usuarios, gracias al cual se podría mejorar la aplicación en aquellos puntos en los que realmente tiene que mejorar. Nadie mejor que el usuario final para saber que está bien y que está mal o falta.



8 Referencias

[1] GLPI. Herramienta para el control de inventario.

Enlace: <http://www.glpi-project.org/>

[2] Scanpet. Herramienta móvil para el control de inventario.

Enlace: <https://play.google.com/store/apps/details?id=com.maiko.scanpet&hl=es>

[3] Enriquez Juan Gabriel, Casas Sandra Isabel, *Usabilidad en aplicaciones móviles* [Online]

Enlace: <http://ict.unpa.edu.ar/files/ICT-UNPA-62-2013.pdf>

[4] Cuotas de mercado de las plataformas móviles. Estadísticas.

Enlace: <http://www.engadget.com/2013/08/01/strategy-analytics-smartphone-share-q2-2013/>
<http://www.engadget.com/2013/10/31/strategy-analytics-q3-2013-phone-share/>

[5] iOS Developer Program. Información.

Enlace: <https://developer.apple.com/programs/ios/>

[6] Uso de librerías Javascript para los sitios web. Estadísticas.

Enlace: http://w3techs.com/technologies/overview/javascript_library/all

[7] Dojo Toolkit. Herramienta de desarrollo.

Enlace: <http://dojotoolkit.org/>

[8] jQuery. Herramienta de desarrollo.

Enlace: <http://jquery.com/>

[9] Sencha Touch. Herramienta de desarrollo.

Enlace: <http://www.sencha.com/products/touch/>

[10] jQuery Mobile. Herramienta de desarrollo.

Enlace: <http://jquerymobile.com/>

[11] Dropbox Sync. Almacenamiento y sincronización de bases de datos en la nube.

Enlace: <https://www.dropbox.com/developers/sync>

[12] Local Storage. Especificación W3.

Enlace: <http://dev.w3.org/html5/webstorage/#the-localstorage-attribute>

[13] Web SQL Database. Especificación W3.

Enlace: <http://dev.w3.org/html5/webdatabase/>

[14] Jerome Etienne QR Code plugin. Homepage.

Enlace: <https://github.com/jeromeetienne/jquery-qrcode>



Análisis, diseño e implementación de un sistema para ayudar en el control de inventario.
Capítulo VIII: Referencias.

[15] Marco Del Corno QR Code plugin. Homepage.

Enlace: <http://thewebthought.blogspot.com/2012/11/jquery-and-javascript-generate-and-read.html>

[16] LazarSoft QR. Homepage.

Enlace: <http://www.webqr.com/>

[17] ZXing QRCode Scanner. Homepage.

Enlace: <https://github.com/zxing/zxing>

[18] Algoritmos MD5 y SHA-1. Librerías.

Enlace: <http://pajhome.org.uk/crypt/md5/index.html>

[19] Ataque basado en colisiones. Información.

Enlace: <http://learncryptography.com/hash-collision-attack/>

[20] Ruptura del algoritmo SHA-1. Información.

Enlace: https://www.schneier.com/blog/archives/2005/02/sha1_broken.html

[21] Métrica Versión 3. Gobierno de España.

Enlace:

http://administracionelectronica.gob.es/pae_Home/pae_Documentacion/pae_Metodolog/pae_Metrica_v3.html

[22] Ley Orgánica 15/1999 [BOE-A-1999-23750].

Enlace: <http://www.boe.es/buscar/doc.php?id=BOE-A-1999-23750>

[23] Verificar checksum en entorno Windows. Información.

Enlace: <http://support.microsoft.com/kb/841290>

[24] Verificar checksum en entorno Mac. Información.

Enlace: <http://support.apple.com/kb/ht1652>

[25] Verificar checksum en entorno Linux. Información.

Enlace: <http://linux.byexamples.com/archives/317/sha-1-checksum/>

[26] Plan de Carrera de Infojobs. Estadísticas de salarios.

Enlace: <http://plandecarrera.infojobs.net/home>



Anexo I: Manual de Usuario



1 Introducción

El objetivo del presente manual de usuario es el de explicar el funcionamiento general de la aplicación móvil iNventariator.

2 Pantalla de login

Al entrar en la aplicación, se encontrará con una pantalla de inicio, en la que aparece un formulario para que introduzca un usuario y una contraseña. Si el usuario existe en la base de datos de la aplicación y la contraseña coincide con la que tiene establecida, se procede al *login*, permitiéndole el acceso a la pantalla principal de la aplicación. En caso contrario, se mostrará un mensaje en pantalla indicándole que el usuario y/o contraseña son incorrectos.



Ilustración 21 - Pantalla de login



3 Pantalla principal

Ésta es la pantalla central, en la que se mostrarán las opciones que pueden ser utilizadas en la aplicación. Son las siguientes:

- Máquinas: opción que engloba todo lo relacionado con el control de las máquinas: añadir, eliminar, modificar, consultar, etc.
- Usuarios: opción que engloba todo lo relacionado con el control de los usuarios: añadir, eliminar, modificar, consultar, etc.
- Log: opción que permite consultar un registro de los últimos cambios que ha habido en la base de datos de máquinas y/o usuarios.
- Administración: opción que permite crear bases de datos de ejemplo, o eliminar las existentes, así como eliminar el log de la base de datos.
- Códigos QR: opción que permite la captura o apertura de imágenes con códigos QR para su decodificación.
- FAQ: opción que permite consultar aquellas preguntas frecuentes sobre el uso de la aplicación.



Ilustración 22 - Pantalla principal de la aplicación



4 Máquinas

En esta opción se engloban todas aquellas acciones que se realizan sobre las máquinas. Está dividida en tres pantallas, a las que se puede acceder a través de la barra horizontal de navegación situada en la parte superior de la pantalla.

4.1 Listado de máquinas

Es la pantalla por defecto que aparece al acceder a la opción Máquinas. En la misma aparecerá un listado con las máquinas existentes en la base de datos. En caso de no existir ninguna máquina, aparecerá un mensaje indicándoselo. Las acciones que puede realizar en dicha pantalla son las siguientes:

- Si pulsa sobre el nombre de la máquina ①, se le redirige a los detalles de dicha máquina.
- Si pulsa sobre el botón con el icono del aspa ②, se le preguntará si desea eliminar dicha máquina de la base de datos. En caso afirmativo, la máquina es eliminada y el listado es mostrado actualizado.

Además, en la parte superior de la pantalla hay un campo ③ que permite la búsqueda instantánea y parcial de máquinas, es decir, los resultados son mostrados a medida de que se escribe, y permite buscar cadenas de caracteres dentro del nombre.



Ilustración 23 - Pantalla Máquinas: Listado



4.1.1 Detalles de una máquina

A esta pantalla se le redirige cuando pulsa sobre una determinada máquina en la pantalla Listado de máquinas dentro de la opción Máquinas. En ella se muestran todos los detalles de la misma, mediante campos expandibles. Los campos nombre y serial aparecen expandidos por defecto, para permitir comprobarlos con mayor comodidad. Puede expandir ⊕ o contraer ⊖ cada uno de los campos pulsando en el símbolo que se encuentra a la izquierda en cada uno de los campos. En la parte inferior de la pantalla puede encontrar dos botones:

- Solicitar: al pulsar en dicho botón la máquina será solicitada, por ejemplo, para sacarla a las oficinas de un cliente, cambiarla de departamento, asignarla a un determinado proyecto, etc.
- Modificar: al pulsar en dicho botón será redirigido a la pantalla de Modificación de máquinas.

The screenshot shows a mobile application interface titled 'iNventariator'. At the top, there are three buttons: 'HOME' with a house icon, 'iNventariator' in the center, and 'Atrás' with a back arrow icon. Below the title bar is a section header 'Detalles de la máquina'. The main content area consists of several expandable fields, each with a minus sign icon on the left and a label: 'Nombre:' (expanded to show 'PCLGN0001'), 'Serial:' (expanded to show 'CCC202122'), 'Modelo:', 'Estado:', 'Tipo:', and 'Situación:' (partially visible at the bottom). Each field has a plus sign icon on the right to expand it.

Ilustración 24 - Pantalla Máquinas: Detalles de la máquina



4.1.2 Modificación de máquinas

En esta pantalla aparecerá un formulario con los valores actuales de la máquina precargados en sus campos. Estos valores pueden ser modificados. Cuando pulsa el botón modificar, los valores que ha cambiado son actualizados en la base de datos y se le redirige a la opción Máquinas con el listado ya actualizado. En caso de que algún campo sea incorrecto, se le notificará.

Ilustración 25 - Pantalla Máquinas: Modificar máquina



4.2 Añadir una nueva máquina

En esta pantalla aparecerá un formulario con todos los datos necesarios para añadir una nueva máquina a la base de datos. Una vez rellenos todos los campos obligatorios y pulsado el botón añadir, la máquina es añadida a la base de datos y se redirige al Listado de máquinas, donde ya se puede ver esta nueva máquina. En caso de que algún campo sea incorrecto, se le notificará.

The screenshot shows the 'iNventariator' application interface. At the top, there is a navigation bar with a 'HOME' button and a title 'iNventariator'. Below this is a tab bar with three options: 'Listado', 'Añadir' (highlighted in blue), and 'Solicitudes'. The main content area contains a form with the following fields:

- Nombre PC:** A text input field.
- Número de Serie:** A text input field.
- Modelo:** A dropdown menu currently showing 'HP Z220'.
- Estado:** A section with three radio button options: 'En Funcionamiento', 'En Reparación', and 'En Stock'.

Ilustración 26 - Pantalla Máquinas: Añadir una máquina



4.3 Máquinas solicitadas

En esta pantalla aparecerá un listado con aquellas máquinas que hayan sido solicitadas por algún usuario. En cada una de las máquinas aparecerán dos opciones:

- Aceptar: la solicitud es aceptada y se inicia el procedimiento para la entrega de la máquina al solicitante.
- Rechazar: la solicitud es rechazada y la máquina es eliminada de la lista de solicitudes.

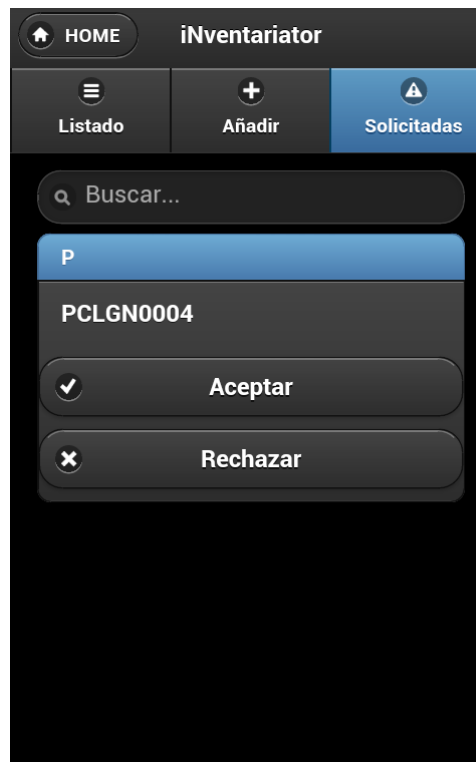


Ilustración 27 - Pantalla Máquinas: Máquinas solicitadas



5 Usuarios

En esta opción se engloban todas aquellas acciones que se realizan sobre los usuarios. Está dividida en dos pantallas a las que se puede acceder a través de la barra horizontal de navegación situada en la parte superior de la pantalla.

5.1 Listado de usuarios

Es la pantalla por defecto que aparece al acceder a la opción Usuarios. En la misma aparecerá un listado con los usuarios existentes en la base de datos. En caso de no existir ningún usuario, aparecerá un mensaje indicándoselo. Las acciones que puede realizar en dicha pantalla son las siguientes:

- Si pulsa sobre el nombre del usuario ①, se le redirige a los detalles de dicho usuario.

Además, en la parte superior de la pantalla hay un campo ② que permite la búsqueda instantánea y parcial de usuarios, tanto por nombre como por primer apellido, segundo apellido o una combinación de los dos anteriores.

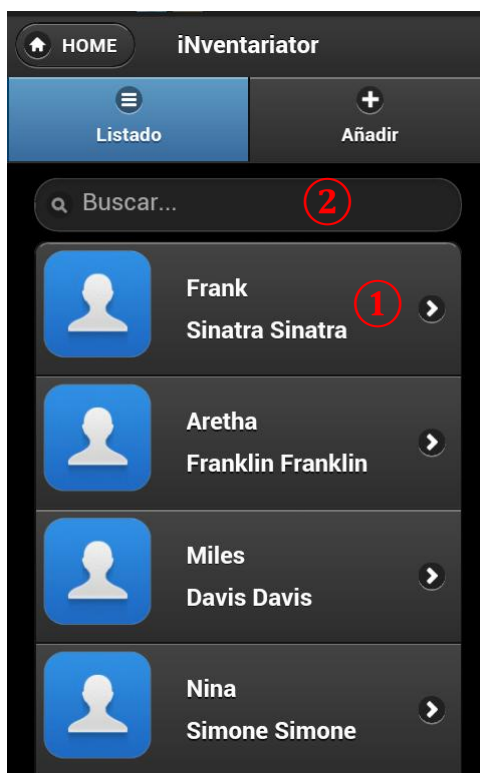


Ilustración 28 - Pantalla Usuarios: Listado de usuarios



5.1.1 Detalles de un usuario

A esta pantalla se le redirige cuando pulsa sobre un determinado usuario en la pantalla Listado de usuarios dentro de la opción Usuarios. En ella se pueden comprobar los datos de contacto del mismo. Estos son: nick, nombre, apellidos, correo electrónico y teléfono. Las acciones que puede realizar en dicha pantalla son las siguientes:

- Si pulsa sobre el botón de eliminar ①, se le pregunta si desea eliminar dicho usuario de la base de datos. En caso afirmativo, el usuario es eliminado y el listado actualizado.
- Si pulsa sobre el botón de modificar ②, se le redirige a la pantalla de Modificación de usuarios.

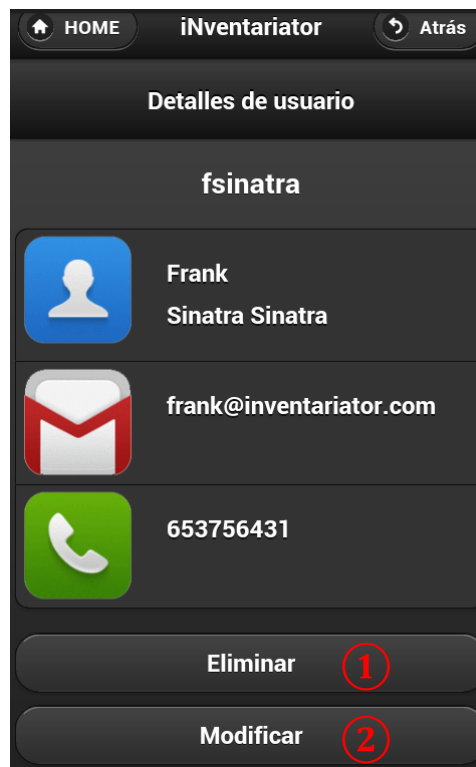


Ilustración 29 - Pantalla Usuarios: Detalles de usuario



5.1.2 Modificación de usuarios

En esta pantalla aparecerá un formulario con los valores actuales del usuario pre-cargados en sus campos. Estos valores pueden ser modificados. Cuando pulsa el botón modificar, los valores que ha cambiado son actualizados en la base de datos y se le redirige a la opción Usuarios con el listado ya actualizado. En caso de que algún campo sea incorrecto, se le notificará.

Ilustración 30 - Pantalla Usuarios: Modificar usuario



5.2 Añadir un nuevo usuario

En esta pantalla aparecerá un formulario con todos los datos necesarios para añadir un nuevo usuario a la base de datos. Una vez rellenos todos los campos obligatorios y pulsado el botón añadir, el usuario es añadido a la base de datos y se redirige al Listado de usuarios, donde ya se puede ver este nuevo usuario. En caso de que algún campo sea incorrecto, se le notificará.

Ilustración 31 - Pantalla Usuarios: Añadir usuario



6 Log de la base de datos

En esta opción aparecerá un registro de todos los cambios que ha habido en la base de datos: usuarios/máquinas añadidos/eliminados/modificados, y los campos que se han visto afectados por estos cambios. Estos registros aparecerán por defecto ordenados de más nuevo a más antiguo. En caso de querer ordenar el log en orden cronológico (de más antiguo a más nuevo) basta con que pulse el botón Invertir Log. En el caso de que no hubiera habido cambios en la base de datos (o si ha eliminado el log), aparecerá un mensaje indicándoselo.

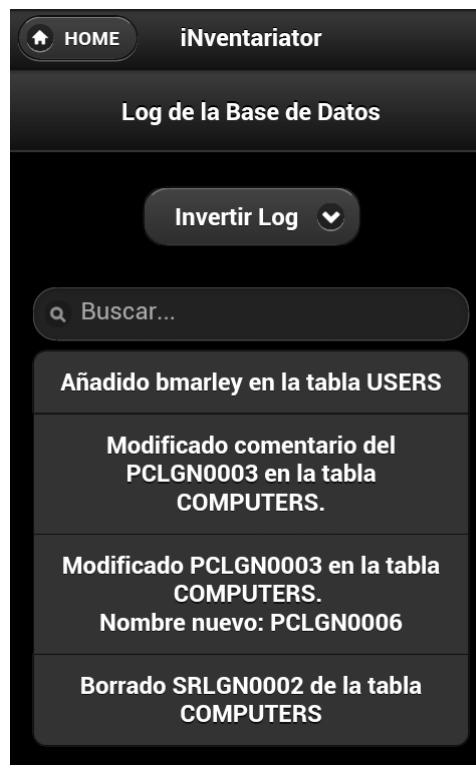


Ilustración 32 - Log de la base de datos



7 Administración

En esta opción se podrán realizar labores de administración de la base de datos. Estas son:

- Crear BD máquinas: se creará una base de datos de prueba con máquinas de ejemplo.
- Borrar BD máquinas: se borrará la base de datos de máquinas, dejándola vacía.
- Crear BD usuarios: se creará una base de datos de prueba con usuarios de ejemplo.
- Borrar BD usuarios: se borrará la base de datos de usuarios, dejándola vacía.
- Borrar Log: se borrará el log generado al realizar cualquier cambio en la base de datos.

Esta pantalla es para uso exclusivo de los administradores.

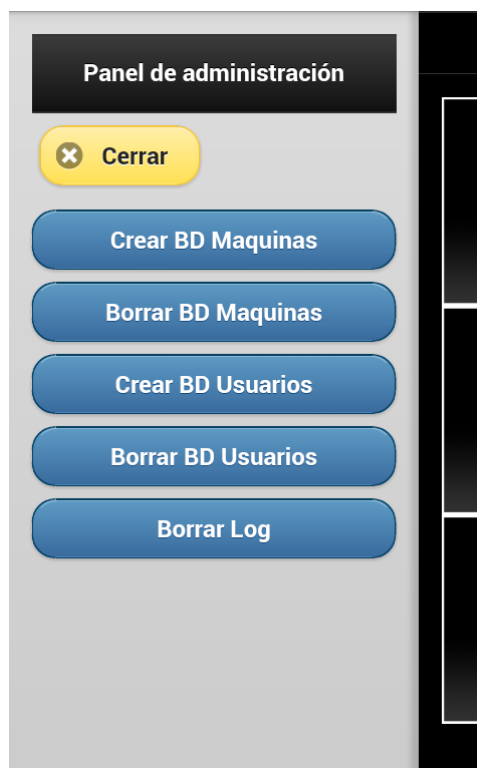


Ilustración 33 - Panel de administración



8 Códigos QR

En esta opción puede decodificar códigos QR. En ella aparecerán dos botones:

- Capturar foto: con esta opción se abrirá la cámara del dispositivo con el fin de poder capturar el código QR. Tras hacer la foto, se decodificará, y en función de si existe ya o no el elemento, se realizará una acción u otra, como se puede ver en la Ilustración 35.
- Seleccionar existente: con esta opción se le permitirá seleccionar una foto existente de su galería de imágenes. Al seleccionar la imagen, se decodificará, y en función de si existe ya o no el elemento, se realizará una acción u otra, como se puede ver en la Ilustración 35.

En ambos casos, se mostrará un mensaje de error en caso de que la decodificación no se realice correctamente.

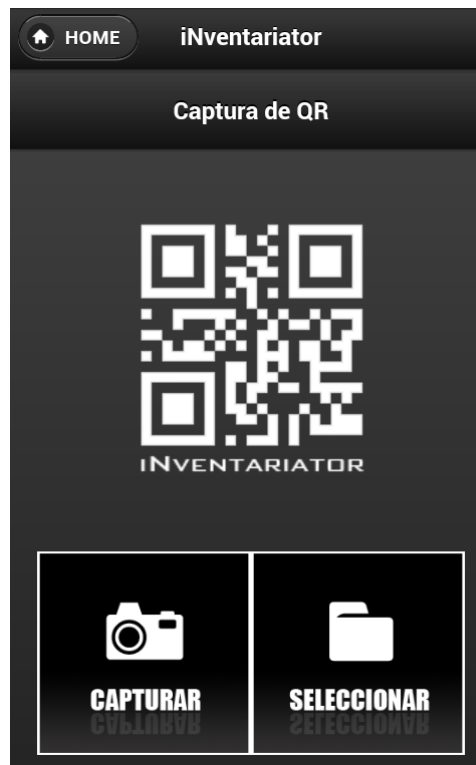


Ilustración 34 - Captura de códigos QR

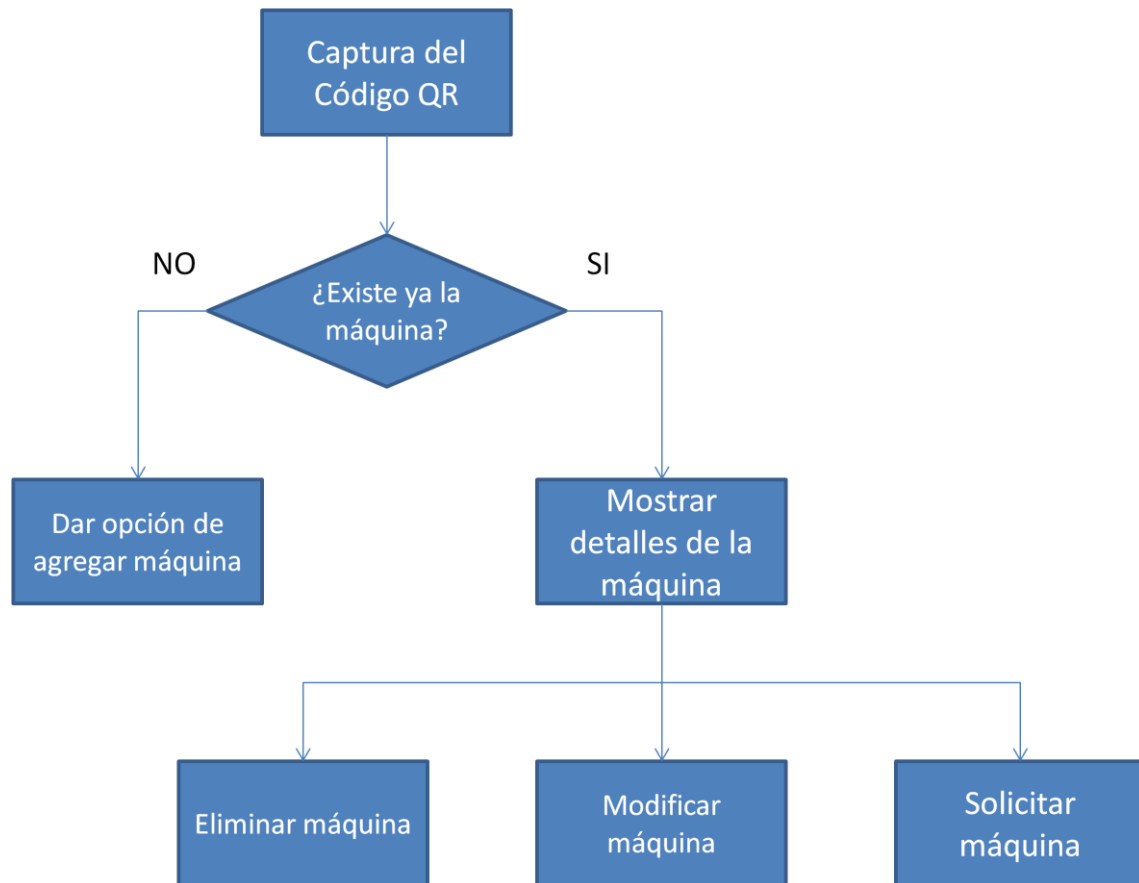


Ilustración 35 - Diagrama escaneo de códigos QR



9 FAQs

En esta opción podrá consultar un conjunto de preguntas frecuentes ya respondidas, por ejemplo: ¿cómo añadir una nueva máquina?, ¿cómo eliminar un usuario?, etc. Tiene la posibilidad de expandir⊕o contraer⊖ cada una de las preguntas pulsando en el símbolo que se encuentra a la izquierda de cada una de ellas. En caso de que ninguna de las FAQs pueda solucionar su duda, en la parte inferior de la pantalla dispone de un apartado de contacto, con el que puede dirigirnos su duda vía Twitter, Facebook o correo electrónico.

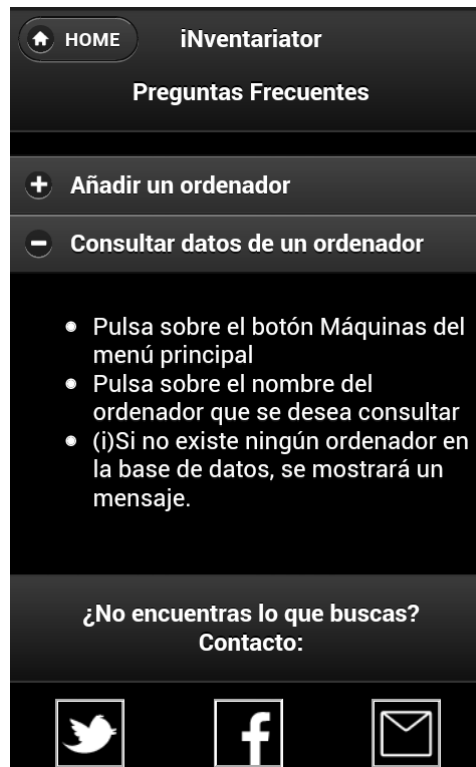


Ilustración 36 - Pantalla de preguntas frecuentes



Anexo II: Informe de pruebas



1 Informes de los casos de prueba

En este apartado se muestran los resultados de las pruebas definidas en la sección Diseño de las pruebas. Los atributos utilizados en los informes de las pruebas son los siguientes:

- **Identificador:** permite identificar inequívocamente a todas y cada una de las pruebas a lo largo del proyecto. Se compone del tipo de prueba (PA para Pruebas de Aceptación) y un número correlativo.
- **Descripción:** contiene la condición que debe satisfacer el sistema para pasar la prueba.
- **Responsable de la prueba:** nombre de la persona responsable verificar el resultado de la prueba.
- **Resultados esperados:** acción que se espera observar al realizar la prueba.
- **Resultados obtenidos:** acción que se obtiene en realidad al realizar la prueba.
- **Resultado:** Pasa la prueba (aceptada) u ocurre algún error (rechazada).
- **Fecha de la realización de la prueba:** fecha en la que se ha realizado la prueba.
- **Firma del responsable:** firma de la persona que es responsable de la prueba, para asegurarse de que se han pasado.

Informe de pruebas		
Identificador	PA-001	
Descripción	Comprobar que el usuario puede loguearse en la aplicación correctamente.	
Responsable de la prueba	Sergio Gómez González	
Resultados esperados	El usuario es autenticado en la aplicación y se le muestra un mensaje de bienvenida.	
Resultados obtenidos	El usuario ha sido autenticado en la aplicación y se le muestra un mensaje de bienvenida.	
Resultado	ACEPTADA <input checked="" type="checkbox"/>	RECHAZADA <input type="checkbox"/>
Fecha de la realización de la prueba	03/06/2014	
Firma del responsable	S.G.G	

Tabla 116 - Informe de prueba PA-001



Informe de pruebas		
Identificador	PA-002	
Descripción	Comprobar que el usuario puede consultar un listado de las máquinas existentes en la base de datos.	
Responsable de la prueba	Sergio Gómez González	
Resultados esperados	Un listado con las máquinas existentes es mostrado por pantalla al usuario.	
Resultados obtenidos	Un listado con las máquinas existentes ha sido mostrado por pantalla al usuario.	
Resultado	ACEPTADA <input checked="" type="checkbox"/>	RECHAZADA <input type="checkbox"/>
Fecha de la realización de la prueba	03/06/2014	
Firma del responsable	S.G.G	

Tabla 117 - Informe de prueba PA-002

Informe de pruebas		
Identificador	PA-003	
Descripción	Comprobar que el usuario puede añadir una nueva máquina a la base de datos.	
Responsable de la prueba	Sergio Gómez González	
Resultados esperados	La máquina es añadida a la base de datos.	
Resultados obtenidos	La máquina ha sido añadida a la base de datos.	
Resultado	ACEPTADA <input checked="" type="checkbox"/>	RECHAZADA <input type="checkbox"/>
Fecha de la realización de la prueba	03/06/2014	
Firma del responsable	S.G.G	

Tabla 118 - Informe de prueba PA-003



Informe de pruebas		
Identificador	PA-004	
Descripción	Comprobar que el usuario puede consultar un listado con aquellas máquinas que han sido solicitadas.	
Responsable de la prueba	Sergio Gómez González	
Resultados esperados	Un listado con las máquinas solicitadas es mostrado por pantalla al usuario.	
Resultados obtenidos	Un listado con las máquinas solicitadas ha sido mostrado por pantalla al usuario.	
Resultado	ACEPTADA <input checked="" type="checkbox"/>	RECHAZADA <input type="checkbox"/>
Fecha de la realización de la prueba	03/06/2014	
Firma del responsable	S.G.G	

Tabla 119 - Informe de prueba PA-004

Informe de pruebas		
Identificador	PA-005	
Descripción	Comprobar que el usuario puede aprobar o rechazar la solicitud de una determinada máquina.	
Responsable de la prueba	Sergio Gómez González	
Resultados esperados	La solicitud es aceptada o rechazada en función de si el usuario la acepta o la rechaza.	
Resultados obtenidos	La solicitud ha sido aceptada o rechazada en función de lo elegido por el usuario.	
Resultado	ACEPTADA <input checked="" type="checkbox"/>	RECHAZADA <input type="checkbox"/>
Fecha de la realización de la prueba	03/06/2014	
Firma del responsable	S.G.G	

Tabla 120 - Informe de prueba PA-005



Informe de pruebas		
Identificador	PA-006	
Descripción	Comprobar que el usuario puede consultar los detalles de una determinada máquina.	
Responsable de la prueba	Sergio Gómez González	
Resultados esperados	Los detalles de la máquina son mostrados por pantalla al usuario.	
Resultados obtenidos	Los detalles de la máquina han sido mostrados por pantalla al usuario.	
Resultado	ACEPTADA <input checked="" type="checkbox"/>	RECHAZADA <input type="checkbox"/>
Fecha de la realización de la prueba	03/06/2014	
Firma del responsable	S.G.G	

Tabla 121 - Informe de prueba PA-006

Informe de pruebas		
Identificador	PA-007	
Descripción	Comprobar que el usuario puede expandir cada uno de los campos de una determinada máquina.	
Responsable de la prueba	Sergio Gómez González	
Resultados esperados	El campo se expande, mostrando su valor.	
Resultados obtenidos	El campo se ha expandido y muestra su valor.	
Resultado	ACEPTADA <input checked="" type="checkbox"/>	RECHAZADA <input type="checkbox"/>
Fecha de la realización de la prueba	03/06/2014	
Firma del responsable	S.G.G	

Tabla 122 - Informe de prueba PA-007



Informe de pruebas		
Identificador	PA-008	
Descripción	Comprobar que el usuario puede eliminar una determinada máquina de la base de datos.	
Responsable de la prueba	Sergio Gómez González	
Resultados esperados	La máquina es eliminada de la base de datos.	
Resultados obtenidos	La máquina ha sido eliminada de la base de datos.	
Resultado	ACEPTADA <input checked="" type="checkbox"/>	RECHAZADA <input type="checkbox"/>
Fecha de la realización de la prueba	03/06/2014	
Firma del responsable	S.G.G	

Tabla 123 - Informe de prueba PA-008

Informe de pruebas		
Identificador	PA-009	
Descripción	Comprobar que el usuario puede solicitar una determinada máquina.	
Responsable de la prueba	Sergio Gómez González	
Resultados esperados	La máquina pasa a la lista de máquinas solicitadas.	
Resultados obtenidos	La máquina ha pasado a la lista de máquinas solicitadas.	
Resultado	ACEPTADA <input checked="" type="checkbox"/>	RECHAZADA <input type="checkbox"/>
Fecha de la realización de la prueba	03/06/2014	
Firma del responsable	S.G.G	

Tabla 124 - Informe de prueba PA-009



Informe de pruebas		
Identificador	PA-010	
Descripción	Comprobar que el usuario puede modificar los datos de una determinada máquina.	
Responsable de la prueba	Sergio Gómez González	
Resultados esperados	Los datos son actualizados en la base de datos.	
Resultados obtenidos	Los datos han sido actualizados en la base de datos.	
Resultado	ACEPTADA <input checked="" type="checkbox"/>	RECHAZADA <input type="checkbox"/>
Fecha de la realización de la prueba	03/06/2014	
Firma del responsable	S.G.G	

Tabla 125 - Informe de prueba PA-010

Informe de pruebas		
Identificador	PA-011	
Descripción	Comprobar que el usuario puede capturar códigos QR mediante la cámara.	
Responsable de la prueba	Sergio Gómez González	
Resultados esperados	El código es capturado por la cámara, y decodificado en caso de que sea correcto.	
Resultados obtenidos	El código ha sido capturado por la cámara, y decodificado en caso de que sea correcto.	
Resultado	ACEPTADA <input checked="" type="checkbox"/>	RECHAZADA <input type="checkbox"/>
Fecha de la realización de la prueba	03/06/2014	
Firma del responsable	S.G.G	

Tabla 126 - Informe de prueba PA-011



Informe de pruebas		
Identificador	PA-012	
Descripción	Comprobar que el usuario puede abrir códigos QR existentes en la galería de imágenes.	
Responsable de la prueba	Sergio Gómez González	
Resultados esperados	El código es abierto desde la galería, y decodificado en caso de que sea correcto.	
Resultados obtenidos	El código ha sido abierto desde la galería, y decodificado en caso de que sea correcto.	
Resultado	ACEPTADA <input checked="" type="checkbox"/>	RECHAZADA <input type="checkbox"/>
Fecha de la realización de la prueba	03/06/2014	
Firma del responsable	S.G.G	

Tabla 127 - Informe de prueba PA-012

Informe de pruebas		
Identificador	PA-013	
Descripción	Comprobar que el usuario puede consultar un listado de los usuarios existentes en la base de datos.	
Responsable de la prueba	Sergio Gómez González	
Resultados esperados	Un listado con los usuarios existentes es mostrado por pantalla.	
Resultados obtenidos	Un listado con los usuarios existentes ha sido mostrado por pantalla.	
Resultado	ACEPTADA <input checked="" type="checkbox"/>	RECHAZADA <input type="checkbox"/>
Fecha de la realización de la prueba	03/06/2014	
Firma del responsable	S.G.G	

Tabla 128 - Informe de prueba PA-013



Informe de pruebas		
Identificador	PA-014	
Descripción	Comprobar que el usuario puede añadir un nuevo usuario a la base de datos.	
Responsable de la prueba	Sergio Gómez González	
Resultados esperados	El usuario es añadido a la base de datos.	
Resultados obtenidos	El usuario ha sido añadido a la base de datos.	
Resultado	ACEPTADA <input checked="" type="checkbox"/>	RECHAZADA <input type="checkbox"/>
Fecha de la realización de la prueba	03/06/2014	
Firma del responsable	S.G.G	

Tabla 129 - Informe de prueba PA-014

Informe de pruebas		
Identificador	PA-015	
Descripción	Comprobar que el usuario puede consultar los detalles de un determinado usuario en la base de datos.	
Responsable de la prueba	Sergio Gómez González	
Resultados esperados	Los detalles del usuario son mostrados por pantalla.	
Resultados obtenidos	Los detalles del usuario han sido mostrados por pantalla.	
Resultado	ACEPTADA <input checked="" type="checkbox"/>	RECHAZADA <input type="checkbox"/>
Fecha de la realización de la prueba	03/06/2014	
Firma del responsable	S.G.G	

Tabla 130 - Informe de prueba PA-015



Informe de pruebas		
Identificador	PA-016	
Descripción	Comprobar que el usuario puede eliminar un determinado usuario de la base de datos.	
Responsable de la prueba	Sergio Gómez González	
Resultados esperados	El usuario es eliminado de la base de datos.	
Resultados obtenidos	El usuario ha sido eliminado de la base de datos.	
Resultado	ACEPTADA <input checked="" type="checkbox"/>	RECHAZADA <input type="checkbox"/>
Fecha de la realización de la prueba	04/06/2014	
Firma del responsable	S.G.G	

Tabla 131 - Informe de prueba PA-016

Informe de pruebas		
Identificador	PA-017	
Descripción	Comprobar que el usuario puede modificar los datos de un determinado usuario.	
Responsable de la prueba	Sergio Gómez González	
Resultados esperados	Los datos de un usuario son modificados en la base de datos.	
Resultados obtenidos	Los datos de un usuario han sido modificados en la base de datos.	
Resultado	ACEPTADA <input checked="" type="checkbox"/>	RECHAZADA <input type="checkbox"/>
Fecha de la realización de la prueba	04/06/2014	
Firma del responsable	S.G.G	

Tabla 132 - Informe de prueba PA-017



Informe de pruebas		
Identificador	PA-018	
Descripción	Comprobar que el usuario puede consultar un log de los cambios que han tenido lugar en la base de datos.	
Responsable de la prueba	Sergio Gómez González	
Resultados esperados	El log es mostrado por pantalla al usuario.	
Resultados obtenidos	El log ha sido mostrado por pantalla al usuario.	
Resultado	ACEPTADA <input checked="" type="checkbox"/>	RECHAZADA <input type="checkbox"/>
Fecha de la realización de la prueba	04/06/2014	
Firma del responsable	S.G.G	

Tabla 133 - Informe de prueba PA-018

Informe de pruebas		
Identificador	PA-019	
Descripción	Comprobar que el usuario pueda invertir el orden en el que es mostrado el log.	
Responsable de la prueba	Sergio Gómez González	
Resultados esperados	El orden en el que se muestran los eventos del log es invertido.	
Resultados obtenidos	El orden en el que se muestran los eventos del log ha sido invertido.	
Resultado	ACEPTADA <input checked="" type="checkbox"/>	RECHAZADA <input type="checkbox"/>
Fecha de la realización de la prueba	04/06/2014	
Firma del responsable	S.G.G	

Tabla 134 - Informe de prueba PA-019



Informe de pruebas		
Identificador	PA-020	
Descripción	Comprobar que el usuario puede borrar el log.	
Responsable de la prueba	Sergio Gómez González	
Resultados esperados	El log se elimina, y aparece un mensaje indicándolo al usuario.	
Resultados obtenidos	El log se ha eliminado, y ha aparecido un mensaje indicándolo al usuario.	
Resultado	ACEPTADA <input checked="" type="checkbox"/>	RECHAZADA <input type="checkbox"/>
Fecha de la realización de la prueba	04/06/2014	
Firma del responsable	S.G.G	

Tabla 135 - Informe de prueba PA-020

Informe de pruebas		
Identificador	PA-021	
Descripción	Comprobar que el usuario puede crear una base de datos con máquinas de ejemplo.	
Responsable de la prueba	Sergio Gómez González	
Resultados esperados	Se crea una base de datos en la que hay máquinas de ejemplo.	
Resultados obtenidos	Ha sido creada una base de datos con máquinas de ejemplo.	
Resultado	ACEPTADA <input checked="" type="checkbox"/>	RECHAZADA <input type="checkbox"/>
Fecha de la realización de la prueba	04/06/2014	
Firma del responsable	S.G.G	

Tabla 136 - Informe de prueba PA-021



Informe de pruebas		
Identificador	PA-022	
Descripción	Comprobar que el usuario puede crear una base de datos con usuarios de ejemplo.	
Responsable de la prueba	Sergio Gómez González	
Resultados esperados	Se crea una base de datos en la que hay usuarios de ejemplo.	
Resultados obtenidos	Ha sido creada una base de datos con usuarios de ejemplo.	
Resultado	ACEPTADA <input checked="" type="checkbox"/>	RECHAZADA <input type="checkbox"/>
Fecha de la realización de la prueba	04/06/2014	
Firma del responsable	S.G.G	

Tabla 137 - Informe de prueba PA-022

Informe de pruebas		
Identificador	PA-023	
Descripción	Comprobar que el usuario puede borrar la base de datos de máquinas.	
Responsable de la prueba	Sergio Gómez González	
Resultados esperados	Se eliminan los datos referentes a las máquinas de la base de datos.	
Resultados obtenidos	Han sido eliminados los datos referentes a las máquinas de la base de datos.	
Resultado	ACEPTADA <input checked="" type="checkbox"/>	RECHAZADA <input type="checkbox"/>
Fecha de la realización de la prueba	04/06/2014	
Firma del responsable	S.G.G	

Tabla 138 - Informe de prueba PA-023



Informe de pruebas		
Identificador	PA-024	
Descripción	Comprobar que el usuario puede borrar la base de datos de usuarios.	
Responsable de la prueba	Sergio Gómez González	
Resultados esperados	Se eliminan los datos referentes a los usuarios de la base de datos.	
Resultados obtenidos	Han sido eliminados los datos referentes a los usuarios de la base de datos.	
Resultado	ACEPTADA <input checked="" type="checkbox"/>	RECHAZADA <input type="checkbox"/>
Fecha de la realización de la prueba	04/06/2014	
Firma del responsable	S.G.G	

Tabla 139 - Informe de prueba PA-024

Informe de pruebas		
Identificador	PA-025	
Descripción	Comprobar que un usuario puede realizar búsquedas parciales e instantáneas en los diferentes listados de la aplicación.	
Responsable de la prueba	Sergio Gómez González	
Resultados esperados	Se muestran los resultados coincidentes con lo introducido en el campo de búsqueda.	
Resultados obtenidos	Han sido mostrados los resultados coincidentes con lo introducido en el campo de búsqueda.	
Resultado	ACEPTADA <input checked="" type="checkbox"/>	RECHAZADA <input type="checkbox"/>
Fecha de la realización de la prueba	04/06/2014	
Firma del responsable	S.G.G	

Tabla 140 - Informe de prueba PA-025



Informe de pruebas		
Identificador	PA-026	
Descripción	Comprobar que no permite loguearse a un usuario inexistente o cuya contraseña no sea correcta.	
Responsable de la prueba	Sergio Gómez González	
Resultados esperados	Aparece un mensaje indicando que el usuario y/o contraseña no son correctos.	
Resultados obtenidos	Ha aparecido un mensaje indicando que el usuario y/o contraseña no son correctos.	
Resultado	ACEPTADA <input checked="" type="checkbox"/>	RECHAZADA <input type="checkbox"/>
Fecha de la realización de la prueba	04/06/2014	
Firma del responsable	S.G.G	

Tabla 141 - Informe de prueba PA-026

Informe de pruebas		
Identificador	PA-027	
Descripción	Comprobar que no permite introducir datos erróneos en los formularios.	
Responsable de la prueba	Sergio Gómez González	
Resultados esperados	Aparece un mensaje indicando el/los campos que no son correctos.	
Resultados obtenidos	Ha aparecido un mensaje indicando el/los campos que no son correctos.	
Resultado	ACEPTADA <input checked="" type="checkbox"/>	RECHAZADA <input type="checkbox"/>
Fecha de la realización de la prueba	04/06/2014	
Firma del responsable	S.G.G	

Tabla 142 - Informe de prueba PA-027



Informe de pruebas		
Identificador	PA-028	
Descripción	Comprobar que el usuario puede desloguearse de la aplicación.	
Responsable de la prueba	Sergio Gómez González	
Resultados esperados	El usuario es redirigido a la pantalla de login.	
Resultados obtenidos	El usuario ha sido redirigido a la pantalla de login.	
Resultado	ACEPTADA <input checked="" type="checkbox"/>	RECHAZADA <input type="checkbox"/>
Fecha de la realización de la prueba	04/06/2014	
Firma del responsable	S.G.G	

Tabla 143 - Informe de prueba PA-028

2 Conclusiones

Como conclusiones extraídas de este informe de pruebas, se puede decir que el *software* desarrollado cumple con los requerimientos del cliente, y por tanto, puede ser entregado al mismo.